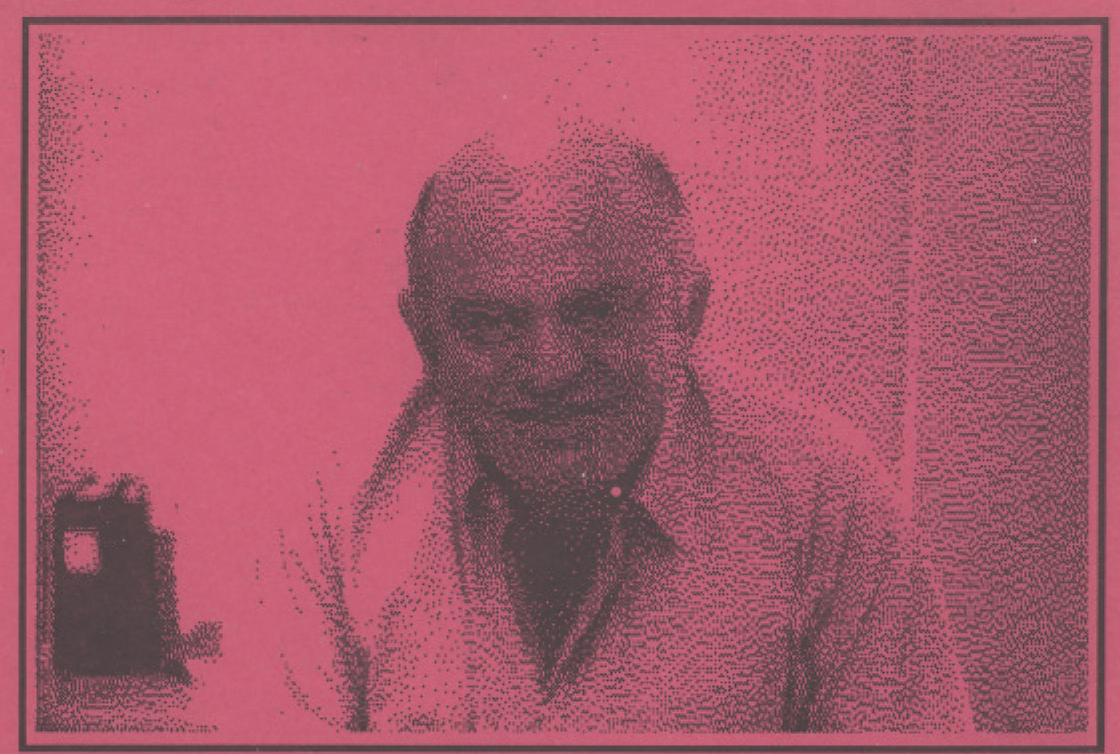
# Bye, folks. I'm off!



# Go ask someone esle

***Luckily there are still some folks supporting your Atari so don't give up***
***Page 6 supports your Atari with the world's oldest and best dedicated magazine ...***
***as well as the world's best PD library ....***
***as well as providing commercial software ...***

**Don't be like the guy above**
**return the favour**

## *Keep supporting Page 6*

*Page 6 Publishing's* NEW

# ATARI USER

*The Resource for the ATARI CLASSIC and the ATARI ST*

Issue 76 - March/April 1996 £2.50

## FOR THE ATARI CLASSIC



- ***THE SOUND SELECTOR***
  ***A superb program that will even write machine code routines for you!***
- ***TEXTPRO MACROS***
  ***A new series to help you get the best from this word processor***
- ***MICRODOS***
  ***Save over 70 sectors on every disk!***

## THE WIDER SCENE

# FREE NETS

*How to use the Internet for nothing!*

Plus a look at the delights of the World Wide Web




*PLUS ... ST PD ROUNDUP ... THE TIPSTER ... MEMORY ROUTINES ... XIO .. JOYSTICKS PORTS ... MORE!*

# This issue's ....

## Thanks

***Les Ellingham*** puts it all together and fills up the gaps but the real thanks goes to the following who made this issue possible

***Sandy Ellingham*** who takes care of all the office work, advertising and mail order

*For their regular contributions*

***John S Davison***
***Paul Rixon***
***Ann O'Driscoll***
***Allan J. Palmer***
***Stuart Murray***

*For their contributions this issue*

***Joel Goodwin***
***David Preston***
***C Ayres***
***Harry Lane***
***John Foskett***
***H S Wood***
***Steve Hooper***
***Andy Guillaume***
***James Mathrick***
***Frank Walters***
***Andrew Pyrski***
***Gordon Hooper***

## APOLOGIES

I am still extremely poor in acknowledging contributions so I apologise to everyone who has sent in stuff and thought it has gone through the wormhole. The intention to reply to everyone is there but the time seems to drift by. If you have not heard, thank you and keep watching the mag, you might be surprised.

## HOW IT'S DONE

PAGE 6 shows just what you can do with your Atari. NEW ATARI USER has always been created entirely with Atari equipment, initially on the XL but more lately with a Mega ST and other stuff, who needs PC's or Macs! Hardware includes a Mega ST2 (upgraded to 4Mb), SM125 Monitor, Supra 30Mb Hard Disk, a HP Laserjet III, Citizen 124D printer, Philips CM8833 monitor, 130XE, a couple of 1050 disk drives, 850 interface, NEC 8023 printer. Principal software used is Protext and Fleet Street Publisher 3.0. Other software includes Kermit, TariTalk, Turbo Basic and various custom written programs on the XL/XE. Articles submitted on XL/XE disks are transferred across to the ST via TARITALK. Programs are coded on the XE and printed out directly for pasting in after the typesetting is completed. All major editing is done with Protext and pages are laid out with Fleet Street Publisher. Each page is output directly from Fleet Street to a HP Laserjet III which produces finished pages exactly as you see them. All that is left is to drop in the listings and photos.

Well, it's not quite as easy as that but you get the idea!

## Inspiration

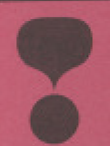
*Well, as I type this, it's Van Morrison that has been on the turntable (is that what they call the inside of a CD player?) for several hours. Before that it was Kathy Mattea for several days. I seem to be stuck in a bit of a rut right now playing the same one or two CDs over and over again. I have tried playing some earlier stuff but somehow can't get the enthusiasm. I guess the withdrawal symptoms last a long time!*

## CONTRIBUTIONS

***Without contributions from its readers, NEW ATARI USER would not be possible. PAGE 6 welcomes and encourages its readers to submit, articles, programs and reviews for publication. Programs must be submitted on disk or cassette, articles should wherever possible be submitted as text files on disk. We seek to encourage your participation and do not have strict rules for submissions. If something interests you, write a program or article and submit it!***

## COPYRIGHT





*Editorial address:* P.O. Box 54, Stafford, ST16 1DR, ENGLAND Tel. 01785 241153
*Editor & Publisher:* Les Ellingham - *Advertising:* Sandy Ellingham
Page layout by PAGE 6 - Printed by Dolphin Press, Fife, Scotland 01592 771652
NEW ATARI USER is published bi-monthly on the last Thursday of the month prior to cover date



**PAGE 6 PUBLISHING's NEW ATARI USER**

**'The Magazine for the Dedicated Atari User'**

ISSN No. 0958-7705

# CONTENTS

**Issue 76 - March/April 1996**

## REGULARS



## ADVERTISERS



## PROGRAMMING



## FEATURES



***The next issue of NEW ATARI USER is due to be published by 31st May 1996***
***Editorial copy date is 30th April***

# SUBSCRIPTIONS

## MAGAZINE ONLY

**Annual subscription rates (6 issues)**

| | |
|---|---|
| UK | £15.00 |
| Europe (Air Mail) | £17.00 |
| Elsewhere (Surface) | £17.00 |
| Elsewhere (Air Mail) | £23.00 |

*Overseas rates reflect only the difference in postal costs*

## DISK SUBSCRIPTION

**A disk containing all of the 8-bit programs from each issue of NEW ATARI USER is available either separately or on subscription. Single price £2.95 per disk, a disk subscription saves you almost £8 a year. Subscription rates (6 issues)**

| | |
|---|---|
| UK | £25.00 |
| Europe | £32.00 |
| Elsewhere (sea) | £32.00 |
| Elsewhere (Air) | £42.00 |

***Please make cheques payable to PAGE 6 PUBLISHING and send to PAGE 6 Publishing, P.O. Box 54, Stafford, ST16 1DR***

# Editorial

Somebody up there doesn't like me. This issue was progressing nicely, back on track and within a day or so of going off to the printers, when disaster struck. I had spent a couple of weeks getting the issue together and had only one or two pages left to do when I returned to find that I could no longer access my hard drive. A couple of hours earlier I had bee using it without problems. I had not even switched it off. It had just decided not to co-operate any more and, no matter what I tried, it would not give up its contents. It was dead. It was no more. It was an ex-drive.

Now I never back up an issue as I go because it takes quite a while and, of course, nothing ever goes wrong does it? Well, I should have recalled five or six years ago when the same thing happened and I had to start all over again. After that I vowed to make regular backups so, although I had lost all of the work on this issue, I knew that most of the drive was backed up. I was horrified to discover, however, that the last backups I made were over a year ago! That means everything that I had done over the past year, that had not been copied on to floppies, was gone. I sat down and made a list of work I could remember and was horrified. There were things like my calendar for the next year, items of stationery, updated customer lists and much more. The biggest disaster (apart from this issue of course) was that I have lost the newly formatted XL/XE PD Catalogue, along with the newly written program that types all the labels for the disks. I still have all the text backed up but it took a great deal of work to get the catalogue into decent shape. Sadly we print out these catalogues on the laser in small batches as they are needed so there may well be delays for those who have ordered a catalogue.

The fortunate side of things is that the subscription database is not affected as that was held on a second hard drive and is, in any case, the only program that is backed up onto floppy disk every time it is updated. It was also fortunate that I had a second hard drive for finances nowadays do not permit splashing out on new hardware, even though prices have come down. It took two days to reformat the old Atari SH204 drive and copy onto it the programs I had backed up a year ago. It took so long because the drive is 10Mb smaller and everything had to be reorganised. First priority was to recreate my Calendar for the next year because I have a number of events booked at weekends which are what pay the bills. I daren't forget one of them! After that I could begin again on Issue 76.

Of course when these things happen, they never happen at the right time do they? This issue I probably typed in (as opposed to transferring from disk) more of the articles than ever before. I had decided to use a number of articles that I had been putting off for some time because they had not been submitted on disk. Not only that, but I had lost a dozen or so articles that had been typed and checked for previous issues and put aside for future use. I had also failed to back up Issue 75 so could not use the usual trick of adapting pages from the last issue. Luckily I had Issue 74 on floppy disk but, even so, there was a lot more work than usual.

Eventually it all got done, although at the time of writing I have only recreated this issue and a few other things and there is a lot more to do. I suspect as the weeks go by I will discover more and more things that I desperately need which I can't recall at the moment. Computers are wonderful aren't they - until they go wrong.

John Davison had a hard disk crash a couple of months ago and this is my second and, of course, we both knew the sense of making regular backups but did we do it? This time I have added a note to my calendar for the 1st of every month to back up the hard drive and I now intend to stick to it. That is, of course, until I think that I will skip one month because I haven't got time. And then, of course, another year will pass by.

*Les Ellingham*



# NEWS AND VIEWS

## SPRING AMS

As many of you will know the Spring AMS will not be held this year at the Bingley Hall in Stafford as the venue was fully booked, but those who need their twice yearly fix of computer bargains can still attend the Spring show. For this year only the SPRING AMS is being held at the Springfields Exhibition Centre in Spalding, Lincs. on Saturday 14th April from 10 am to 4 pm

Neither ourselves not Derek Fern are attending but at the time of writing it is known that TWAUG are going along and there may well be other Atari support. Of course all the usual bargains in consumables and second hand equipment and the like will still be on offer, so it should still be an interesting day out.

The Springfields Exhibition Centre is a well known venue in the East of England but if you want more details of the event, give Sharward a ring on 01473 741533.

AMS will be back at the Bingley Hall in November this year and we'll bring you more information on that show nearer the time.

## ATARI NO MORE?

A recent report in the trade paper CTW states that Jack Tramiel has spent some $30 million from Atari's coffers to buy shares in a Californian hard disk drive manufacturer. Although no comment was forthcoming from Atari themsel-ves, the report states that the move is a 'merger' and that the hard disk company will have the controlling interest with Atari as merely a retailing arm of the company.

The Tramiel's have said that this move will enable them to recoup on their investment for the development of future technologies but have refused to say whether any money will be used for support of the Jaguar or any computer related product. Sounds to us as though this is the end of Atari with Jack Tramiel realising that he'll end up with nothing if Atari go on their merry way. In the third quarter of 1995 Atari's turnover was down to around $4 million but it cost them $17 million to trade during that period, thus making a substantial loss. Looks like Big Jack's interest in computers has finally waned but we all knew that some time ago, didn't we?

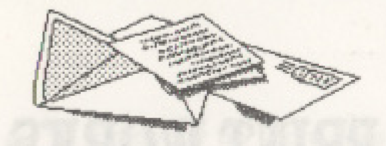
## PRINT WORKS

New from Micro Discount is PRINT WORKS, described on the packaging and disk as a Document Processor For The Atari 1029 Printer. We have had a copy for review for some months but haven't brought you a review because we no longer have access to a 1029 printer.

I promised that we would mention the product as a news item this issue but on looking through the package find that it seems to work on any other printer as well, with a facility for creating specific printer drivers. If only we had known! The package looks to be an excel-lent program for formatting documents, a sort of cross between a word processor and a full DTP program, and should be of interest to any-one with a printer.

If we can get the program running on either a Citizen 124D or the NEC 8023 then we'll bring you a review next issue. In the meantime if you want further details give Micro Discount a ring on 021 353 5730.

# Mailbag



## Whither art thou?

*Many correspondents seem to have started 1996 off in a state of lethargy as we have had very few letters for Mailbag this time. I kept on hanging on before sending them off to Allan Palmer, only to find that I had really left it too late for him to be able to get a column together on time. I apologise for that and am thankful that a few letters came in at the last minute to give us a reasonable column. For the next issue, let's get things going again so that I can send all the correspondence off to Allan for him to put together his usual interesting column. Write now!*

***Les Ellingham***

## AMENDMENT

A small error crept into the printed listing in last issue's Mailbag for the MARGIN PRINT program by H.S. Wood. Line 29130 should read:

```
29130 G=USR(C):RESTORE 0:
END
```

I am sure that many of you spotted this, but thanks to Mr Wood for bothering to check his program.

## SOUND SUPPORT

*James Mathrick from Rowlands Castle in Hants writes:*

I would like to agree and support the points raised in the Editorial last issue - I for one do not mind Page 6 issues being delayed if it means the high standard of the magazine is maintained. I find this magazine extremely helpful in a world of dwindling support for the Atari, and it provides a useful link between users. I am glad that Page 6 consider it worthwhile continuing to support the Atari, along with the advertisers, many of which I know gain no profit for this support but do so solely for the love of their machines.

Many thanks for publishing my tips in The Tipster last issue - I hope they keep The Tipster column going for a few issues. I have recently received some more commercial software and should have some more tips soon. Thanks also to Andy Guillaume for his excellent '256 colours' routine. I had been working away on this for several months with little real success, as I only started machine code programming in the summer of '95, so many thanks to Andy for saving me many weeks of frustrating work. I am well into a paint program using these colours, which should lead onto some decent programs - watch this space!

I have had some problems with Alternate Reality - The Dungeon as the game refuses to accept Disk 3 Side 2 (Side 1 is accepted) and as the disk contains the D&P and other necessary establishments I would like to rectify the problem. I suspect some disk damage and I would be grateful to hear from anyone who has an original master disk 3 and can supply a backup copy of this. I will gladly supply the cost of the blank disk.

I have recently been finding out about sound digitising and, after some contact with Dean Garraghty (who was exceedingly helpful), I have constructed the Antic Sound Processor (from an early Antic magazine) and am impressed by the possibilities it opens up, however I would like to improve on the quality of the sampling. In Dean's words it sounds as if someone is hoovering in the background. Dean mentioned that fitting an external ADC (analogue-to-digital convertor) would help, but my problem is this. Most ADCs supply the digital information in 12 or 8 bits and the volume control of the Atari is only 4 bits wide. I would like to reach a standard of 2-bit Systems Replay, but as I understand it these units are few and far between. If anyone has Replay and has a better understanding of electronics than me, and can advise me as to how to improve the sampling quality, I would be most grateful. Also I would like to maintain screen display while playing back a sound sample but this heavily distorts the sound, quite how I know not. I assumed it slowed down the playback data feed, but increasing the play rate does not affect the distortion.

Lastly a couple of quick queries. First, how can graphics modes be changed in machine code? I have played around with location 623 but this does not work. Secondly, how can tasks be performed while loading data/programs, as in the loading routines in Hardsoft's 'Cool Emotion'? As I understand it, loading from disk is interrupt driven, so do new interrupt routines need to be created in order to play music etc.?

Sorry to bombard you with so many questions and not with helpful tips and articles. Good luck with the magazine, I hope it continues long into the future. I will certainly support it for as long as I can.

*Many thanks for all your words of support James, it really does help when things are slow and not much post is coming in. Your letter gives the opportunity to other readers to start flooding in letters of help as you raise a number of points that other readers surely know the answers to. I suspect that some of the sound sampling problems may be rather complex but I am sure that someone has at least a few answers. And your last few queries should get a response from the dedicated machine code readers out there. If you have any answers for James, write in and we will share them with other readers in next issue's Mailbag. If you have an Alternate Reality disk or want to get in touch with James direct let us know and we will pass on your name and address.*

## FROM VCS TO HYPERCOPY

*Some reminiscences and an offer of help come from Jack Worton in Ironville, Notts:*

I cannot remember when I first became attached to the Atari, it was so very long ago. I purchased a VCS system from The Byte Shop in Nottingham for my children, but there are no prizes for guessing who was on it the most. At the time warranty cards had to be returned to 'Cherry Leisure' and I was wondering if these were the sole agents at the time. Games were then very scarce and when they did appear the bank manager had to be with you to decide if a loan was appropriate!

I had to travel as far as Sheffield for the next game and the prices I paid for them was anything from £20 to £45. And I'm talking about the seventies. The 400 and 800 were unheard of then. I was very fortunate when they did come out as the job I had paid a decent wage. Because of this I did some work for a local businessman free as we were also good friends. One day he sent for me and I went, fully tooled up, and ready for the job in hand but was greatly surprised when he dropped this rather big



parcel in my hands and said it was mine. You cannot imagine the joy I felt when I discovered what was inside - an Atari 800, fully boarded for maximum RAM and also he had put in some of the books that were available then to assist with learning. There had to be a problem, but this did not show for quite a while when I was typing in programs out of the books. I had no means of saving them. I eventually got my tape recorder which made me happier but the day I got my first disk drive took me over the moon. It cost over £300 but it proved to be invaluable.

Over the years I got in with the Atari clan and was soon on the way to owning all the things one shouldn't at the time like those things you mention in your write up about the HYPERDRIVE. Things like Happy's, Archivers and Duplicators and soon I was becoming quite adept at copying protected software. Taking it out was quite beyond me.

Now I am quite alone, but still thoroughly enjoy my time on the old faithful. It's a 130XE now but the 800 is just a few feet away. I'll have to plug it in and give it a whirl before the caps go dry. With no-one around these days I have to back up my own software and as a result have had to study the protection methods in order to be able to write out the nasty little routines to make them copyable. I do not do this on a scale greater than one, me, normally but now I have read your latest review I get the feeling that there are some out there less fortunate than myself who cannot handle basic Assembly Language. I say basic because I am no expert. I was wondering if I could maybe offer my services to anyone who is experiencing difficulty backing up his or her favourite disks. I know what it is like losing originals. I once had a disk drive that was a pig at scouring tracks and I never could find out why.

I do not know of any disks I cannot back up now either by copying it or tracing the protection and eliminating it. At the moment BANK BANG is going through the process. I don't know if it is just my copy that is faulty or whether the drip who protected it really meant to damage or wear my disk drive out. The program asks the drive to read all the last track almost and it isn't even formatted. My read/write head does forty-two trips from track zero to forty. Is this protection for real or is it my copy? It's worse than loading a tape.

Anyway if I can be of help then you may spread my name around, if not then simply file me under 'BIN'. I shall always look forward to my next copy of NAU and I will never complain about it being late as others seem to do.

❢ *Thanks for the reminiscences Jack, I am sure they stirred a few memories with many of us. Your remarks about backing up software will still raise a few hackles in some quarters, as I know the folks who produce the current games like Bank Bang are quite paranoid about piracy. I am sure that all the gnashing of the disk drive heads is intentional and not just on your copy but I have always hated those programs that are so heavily protected that they either grind away on the drive or cease to load when the drive goes a little out of spec. If anyone wants to take advantage of Jack's offer to help in backing up software, I will pass on your letters and requests but, of course, don't be foolish enough to start spreading around copies of software that is still available. There is still software to be had from the likes of Micro Discount and Dean Garraghty and we want to hang on to those suppliers as long as possible.*



## KEEP IT ON PAPER

*Chris Thorley up in Elgin is a bit worried that NAU might disappear onto a disk:*

Much discussion has recently focused on the format of Page 6 and even as to whether a printed magazine is what the readership wants.

Well I for one prefer a printed magazine to a disk based one. In fact I would gladly pay an increased subscription if necessary. With even today's modern, hi-tech computers, magazine sales play a huge part in the promotion of computing as a whole.

Page 6 may not be the glossy magazine it once was but I am still grateful for the effort put in by Les, Sandy and all the contributors in producing an information packed magazine, which helps to keep us all in touch with the Atari Classic world.

❢ *There are no proposals to do away with the printed page of Page 6, despite what you may have read elsewhere. I have to agree with you that the printed page is still, by far, the most effective medium for communicating and sharing thoughts and ideas. Although the Internet seems to be growing at an enormous speed, the majority of people still communicate via the printed word and I am sure that it will continue thus for many years, certainly into the middle of the next century. Those who say that the electronic page will see the end of the printed page are giving too subjective a view and fail to realise that there are millions of people who do not own a computer and millions who will never do so. They fail to realise that anyone can afford 25p for a newspaper, or can even pick up one someone has left behind, but it is still the privileged minority who can afford to invest £1,000 plus on a system that will enable them to contact the world. Electronic communication will grow to heights undreamt of but the printed page will survive as long. There is still something about holding the printed word in your hand that no electronic means can replace. Isn't it strange that you can buy magazines that are dedicated to the Internet?*

## CRAZY ABOUT IT

*Paul Kovacs in Darlington is suffering from a common ailment that affects Atari Classic users:*

Being a long time supporter of the Atari Classic has turned me into a kind of Atari fruitcake to put it mildly. While all my friends have abandoned their 8-bit roots and keep going on about their 486 PC's with multimedia this and that, I quietly go about my way hunting out all those Atari bits and bobs. Over the last 10 years or so, I have accumulated a large pile of Atari software, books, mags, and lots of hardware. The computers alone, at the last count, are an Atari 400, an 800XL, two 130XE's, one with a switchable OS installed, and an Atari STFM which I upgraded to 4 meg memory and added TOS 2.06 OS. At the moment I now have two hard drives, one is a 640Mb and the other is a 40Mb drive.

I would like to know if it is possible to connect a hard drive to the 130XE, and if it is do you need some kind of interface for it. Also what software would you need to use the hard drive with the 130XE? I am sure that over the years someone must have had some thoughts about this kind of thing. I have looked through all my mags and books to see if somebody has done this before but have found nothing on this subject. I hope that you or any readers could shed some light on the subject.

❢ *It's all been done before, Paul! Supra developed a hard drive interface for the XL/XE*



*some yearsago and it was distributed in this country by Frontier Software as a complete unit. We even carried a review in issue 29 and I can remember the excitement of having a hard drive hooked up to my 130XE in the office. Sadly the idea of the hard drive based 8-bit system never took off, primarily because the prices asked were horrendous - would you believe £750! - and Frontier only sold a couple of drives in this country. There were many more sold in the States and quite a few are still in operation but there is not a great deal mentioned about them any more. I seem to remember mention of a separate interface at some stage but I don't think anything came of it. As things stand, you are unlikely to be able to use your hard drive with your 130XE, even though it is quite possible, unless someone knows where there is a spare interface lying around.*

## TYPE-IN TROUBLES

*A welcome letter comes from John Foskett with advice on programs he has had published and news of more to come:*

Regarding the typing problems expressed in Mailbag of Issue 75 about the problems with typing in the machine code strings in my Lottery and CES programs, firstly I apologise for the trouble they have caused.

The advantage in using directly defined strings is that there is practically no initialising time compared to loading data via a loop. Regarding my CES program, the earlier version published way back in issue 63 contained data which took about 5 seconds to load and I decided to eliminate this with CES version III. Eliminating the initialising time would be a great advantage because CES is designed for people to use within their own programs and its initialising time would obviously be added to that of the host program.

I'm afraid that I am guilty of taking machine code strings for granted because I never need to type them in. After writing the source code for the routines and assembling them, I simply convert the code directly into strings and, hey-presto, machine code strings without any trouble.

Because of the advantages of strings, I will continue to use them in my programs, but in the light of the typing difficulties, I will from now on also write a special routine which will write the strings for you from a bank of numeric data. When run, the routine will convert the data into strings and write them to disk so that it will be a simple matter of typing NEW, ENTERing the strings and typing the rest of the listing around them.

*John also supplied a breakdown of the problem lines (line 570 in the Lottery program and line 120 in CES) which we will happily pass on to any readers still having problems. We have already received more contributions from John, featuring his string creating routines which will help with type-ins. Look out for some great programs in this and future issues. Thanks, John.*

## TRANSDISK

*Finally, Brad Rogers from way down on the South coast, in Southampton, has a little help with some Transdisk problems:*

In Issue 74, Les asked readers if they could give information about getting various tape based games on to disk using Transdisk IV. The only one listed that I can help with is Universal Hero. To transfer set up as follows: XE Menu, Non-standard load. No Segment numbers are required. It should be noted that there is a bug in Universal Hero that makes it impossible to complete the game.

It was nice to see The Citadel as the Bonus program, especially since I know the author, albeit purely via e-mail.

*Brad also asked some questions about the late publication of Issue 74 which I hope were answered by last issue's editorial. The info about Universal Hero is gratefully received but surely this can't be all the help we are going to get with Transdisk? Surely someone has tried to transfer all those games so let us know how you got on, even if it is to say that it can't be done.*

*Well, that's the end of Mailbag this time, rescued by a couple of fairly long letters. I am sure that there is a lot more to be discussed, shared and enthused upon in these pages so get your pen to paper, or fingers to the keyboard, and get writing soon so that I can get your letters to Allan to create his usual interesting column next time. As you will see my style is somewhat different and I am sure you prefer Allan's, so let's have lots of correspondence once again.*



**It's nae good phoning laddie, ye'll ha' to put it in writing**

## WRITE TO US!

***Air your views on all things Atari or help your fellow users with their queries - even ask for help yourself if you want. It's all interesting, if only you write it down. Here's the address:***

**MAILBAG**
**NEW ATARI USER**
**P.O. BOX 54, STAFFORD**
**ST16 1TB**



# M-EDIT

## *Some useful machine code routines from Joel Goodwin*

Two very useful machine language routines are clear memory and move memory routines - they can be considered to be "memory editing" routines, hence the title of this article. Recently I have been looking at ways of tightening up the coding of these routines and believe I have developed extremely efficient versions of them.

## THE ROUTINES

To use the routines you must include the M-EDIT initialisation subroutine in your own program. This must be called early on in your program using a 'GOSUB 8500' command. Note that your program can only use page 6 memory locations 1641-1791 as M-EDIT takes the rest for itself. Also do not put anything into memory locations 203-204 because any data here will be lost when M-EDIT is called.

How do you use them? Well, to clear N bytes of memory starting from location ADDR you would use:

**100 Z=USR(CLEAR,ADDR,N)**

For example to clear a graphics 0 screen:

**100 Z=USR(CLEAR, PEEK(88) + PEEK(89)*256,960)**

To copy N bytes of memory starting from location FROM to the area of memory starting at location TO, you would enter:

**100 Z=USR(MOVE,FROM,TO,N)**

For example, to copy the Atari character set into a string CH$:

**100 Z=USR(MOVE,57344, ADR(CH$), 1024)**

Important point - when moving memory the memory block you are copying from should not overlap the block you are copying to.

## A DEMONSTRATION

It is usual to animate a full screen of graphics using a neat technique such as page flipping. However, to demonstrate the speed of the move routine, the demo program animates by copying 6400 bytes of graphics data to screen memory each time a new frame of animation is to be displayed.



```
LD 8494 REM           M-EDIT
VE 8495 REM        Joel Goodwin
TD 8496 REM  NEW ATARI USER - March '96
PY 8497 REM  INITIALISE M-EDIT ROUTINES
OV 8498 REM Page 6 locations 1641-1791
MG 8499 REM are still free for your use
CE 8500 RESTORE 8600:FOR M=1536 TO 1640:R
   EAD MB:POKE M,MB:NEXT M
PF 8510 CLEAR=1536:MOVE=1582:RETURN
BD 8600 DATA 104,104,141,30,6,104,141,29,
   6,104,73,255,133,204,104,73,255,133,20
   3,230
SG 8610 DATA 203,208,2,230,204,169,0,168,
   153,255,255,200,208,3,238,30,6,230,203
   ,208
FI 8620 DATA 243,230,204,208,239,96,104,1
   04,141,83,6,104,141,82,6,104,141,86,6,
   104
IL 8630 DATA 141,85,6,104,73,255,133,204,
   104,73,255,133,203,230,203,208,2,230,2
   04,160
IF 8640 DATA 0,185,255,255,153,255,255,20
   0,208,6,238,83,6,238,86,6,230,203,208,
   237
MR 8650 DATA 230,204,208,233,96
YX 8660 REM * 105 BYTES
```

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

Although the animation is not smooth, and the program only cycles through three frames of animation, it does show how fast the move routine is.

## GETTING TECHNICAL

So what makes these routines more efficient? Well there are two key ideas involved and I'm going to start talking about some machine code here so skip this if it's not your scene.

```
EX 1 REM ################################
TK 2 REM #        M-EDIT DEMO           #
PX 3 REM #      by Joel Goodwin         #
HB 4 REM # ---------------------------- #
DV 5 REM # NEW ATARI USER -MARCH '96    #
FC 6 REM ################################
NM 7 REM
RA 30 GOSUB 8500
EB 40 GRAPHICS 15:DIM P1$(6400),P2$(6400)
   ,P3$(6400),P(3)
UD 50 P1$(1)="A":P1$(6400)="A":P1$(2)=P1$
   :P2$=P1$:P3$=P1$:P(1)=ADR(P1$):P(2)=AD
   R(P2$):P(3)=ADR(P3$)
GL 60 SC=PEEK(88)+PEEK(89)*256:POKE 708,1
   2:POKE 709,8:POKE 710,4:POKE 752,1
OJ 70 FOR OFF=1 TO 3:GOSUB 200
VX 80 X=USR(MOVE,SC,P(OFF),6400)
EA 90 X=USR(CLEAR,SC,6400):NEXT OFF
EY 100 GRAPHICS 63:POKE 708,12:POKE 709,8
   :POKE 710,4
ZL 110 FOR A=1 TO 3:X=USR(MOVE,P(A),SC,64
   00):NEXT A:GOTO 110
SI 200 C=OFF:? "[ESC,CLEAR][ESC,DOWN]
   [ESC,TAB][ESC,TAB]Screen ";OFF:FOR A=0
   TO 79
GY 210 C=1+C*(C<3):COLOR C
AT 220 PLOT 79-A,79-A:DRAWTO 80+A,79-A:DR
   AWTO 80+A,80+A:DRAWTO 79-A,80+A:DRAWTO
    79-A,79-A
JM 230 NEXT A:RETURN
```

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

***Listings - the M-EDIT routine (listing 1) should be saved in LIST format to be ENTERed into your program. The DEMO program above should be SAVEd and then the M-EDIT routine ENTERed before being SAVEd again. The demo will not work without the first program being added***

If you look at the assembly language listings of both the MOVE and CLEAR routines you will notice that NUM, the number of bytes to be cleared/moved, is manipulated before being used as a counter. I actually use 65536-NUM as a counter. Why is this? The trouble with NUM is that you end up with a code very similar to this:

```
LDX NUM
DEX
STX NUM
CPX #$FF
BNE NEXTBYTE
DEC NUM+1
```

There is a CMP statement for every byte cleared/moved. By using C=65536-NUM I can count upwards instead which means I can do this instead:

```
INC C
BNE NEXTBYTE
INC C+1
```

Not only have I removed the X register from the proceedings, but it is now far easier to check whether the high byte of the counter is changing. The edit routine will stop once we have counted C up to zero.

The second key idea involves self-modifying code. Self-modifying code is usually dismissed as bad programming. It can be difficult to debug and automatically forces your code to be non-relocatable. As we are discussing a pair of small routines then I think in this instance we can justify such an approach.

Normally in a move routine, you would code something like the following:

```
        LDA (FROM),Y ;Y register is zero
        STA (TO),Y
        INC FROM
        BNE CHECKTO
        INC FROM+1
CHECKTO INC TO
        BNE NEXTBIT
        INC TO+1
NEXTBIT ........
```

Those INC instructions are 5 machine cycles each. We can make a pretty good saving to begin with by re-programming this as:

```
        LDA (FROM),Y
        STA (TO),Y
        INY
        BNE NEXTBIT
        INC FROM+1
        INC TO+1
NEXTBIT ........
```

That's pretty good, because the two main INC's have been replaced with a single INY which saves 8 machine cycles per copied byte, and that is not considering the branch instruction that has disappeared. Now if we accept self-modifying code we can make another good saving:

```
XFROM   LDA $FFFF,Y
XTO     STA $FFFF,Y
        INY
        BNE NEXTBIT
        INC XFROM+2
        INC XTO+2
NEXTBIT ........
```

What we do here is write directly to the instructions. Our pointers to the bytes being copied from and to are now the instructions themselves. I put $FFFF because prior to copying we would have loaded the proper addresses in (see the assembly listings). This has saved us an extra 2 machine cycles per byte copied. Although the INC/INY tactic saved more, this is important because it is an uncommon programming strategy hence why



```
        *= $0600
NUM     =  $CB
;
;  MCLEAR Memory clear
;  Input: ADDR,NUM
;
MCLEAR  PLA             ;Discard
        PLA             ;Get ADDR
        STA MC0+2
        PLA
        STA MC0+1
        PLA             ;Get NUM and put
        EOR #$FF        ;NUM=65535-NUM
        STA NUM+1
        PLA
        EOR #$FF
        STA NUM
        INC NUM         ;Add 1 so that
        BNE MC          ;we get 65536-NUM
        INC NUM+1
MC      LDA #$00
        TAY
MC0     STA $FFFF,Y     ;Store zero
        INY             ;Y=Y+1
        BNE MC1
        INC MC0+2       ;if Y=0 then
                        addrhi=addrhi+1
MC1     INC NUM         ;numlo=numlo+1
        BNE MC0
        INC NUM+1       ;if 0 then
                        numhi=numhi+1
        BNE MC0
        RTS             ;if that's 0 then
                        done

;  MMOVE Memory Move
;  Input: FROM, TO, NUM
;
MMOVE   PLA             ;Discard
        PLA             ;Get FROM
        STA MM0+2
        PLA
        STA MM0+1
        PLA             ;Get TO
        STA MM1+2
        PLA
        STA MM1+1
        PLA             ;Get NUM and put
        EOR #$FF        ;NUM=65535-NUM
        STA NUM+1
        PLA
        EOR #$FF
        STA NUM
        INC NUM         ;Add 1 so that we get
        BNE MM          ;NUM=65536-NUM
        INC NUM+1
MM      LDY #$00
MM0     LDA $FFFF,Y     ;Copy data
MM1     STA $FFFF,Y
        INY             ;Y=Y+1
        BNE MM2
        INC MM0+2       ;if Y=0 then
                        fromhi=fromhi+1
        INC MM1+2       ;and tohi=tohi+1
MM2     INC NUM         ;numlo=numlo+1
        BNE MM0
        INC NUM+1       ;if 0 then
                        numhi=numhi+1
        BNE MM0
        RTS             ; if that's 0 then done
```

**MEMORY EDITING ROUTINES by Joel Goodwin 10-6-95. Keypoints: Self-altering code; ; Convert "number of bytes" into a counter that goes up rather than down**

I have declared it as a key idea.

My editing routines are longer than other routines but this is one of the small sacrifices that has to be made in exchange for the efficiency.

## ROUNDING OFF

The speed of the routines is not usually of importance in BASIC, because just about anything in machine code is far better than BASIC's speed. These routines are of use to BASIC programmers but the efficiency will be more important to machine code programmers looking for ways to speed up time-expensive code.

The real aim of this article is to encourage more programmers, from whatever language background, to seek out better solutions to programming problems. More often than not, there is a either a cleaner or quicker piece of code than the one you first came up with. ●

# PORT WITHOUT THE WHINE

***Steve Hooper, who also came up with the astounding title, presents a quick tutorial on how to program the joystick ports***

The XL/XE machines have a wide variety of ports used by the Operating System for input and output of data and control signals. The Parallel Bus and Serial Port are probably only programmable for certain uses if you have an acute understanding of machine code and the monitor and television jacks aren't exactly user serviceable. This leaves just the joystick (or controller) ports for input and output.

Input is easy, of course. We can obtain the status of any input device, such as a joystick or paddle, simply by using the BASIC command ? STICK(0) and ? PADDLE (0) respectively. Using the ports for output, however, is a little more difficult. The BASIC language doesn't allow commands such as STICK(0)=13. Fortunately though, we don't have to bid an unwelcome return to hexadecimal machine code to be able to program the ports. All that is required are a few POKE instructions.

Normally, the joystick pins act as input connectors allowing completion of circuits within the computer. Your stick, in all its glorification (or not in the case of the CX10), merely consists of a selection of switches, one or more of which are closed when the joystick handle is pushed in a particular direction. There are five switches in the average joystick corresponding to the Forward, Back, Left and Right directions with the fifth one for the Fire button. Circuits are completed between these five +5volt lines and a single common Ground line (0volt).

## LET'S RECONFIGURE

The Atari machines contain one or two Peripheral Interface Adapter (PIA) chips - the 400/800 models contain two. This chip is responsible for configuring the joystick ports to allow output instead of input. With just a few POKEs we can gain control over this section of PIA. Type the following (in BASIC!) and press Return.

**POKE 54018,34**

Having done this, all direction lines will be a 0v instead of the normal +5v level, in both ports 1 and 2. It should be noted that the trigger lines remain unchanged, still as input, as these are connected to GTIA (or CTIA in the case of 400/800 models) and not to PIA. This is useful, since they can be used for simple sensing if the device requires this.

In order to switch or 'send' a current to one or more lines in the ports, we must make use of another POKE command. This takes the form of

**POKE 54020,X**

What numeric value should take the place of X? Take a look at Figure 1 and all will become clear. For example, if we POKE 54020,13 the 'Back' line (number 2) is activated and switched from 0v to +5v. In order to control the lines in port two as well, the following formula must be used:

**POKE 54020, X1+(16*X2)**

where X1 and X2 refer to ports one and two respectively. So POKE 54020,238 would change the 'Front' direction lines of both ports to a high (+5v) level, because 14+(16*14)=238. Simple, eh?

When you want to return both ports for input type

**POKE 54020,255: POKE 54018,31**

**PORT #1**

| Value | BACK | FRONT | LEFT | RIGHT |
|---|---|---|---|---|
| 0 | ✣ | ✣ | ✣ | ✣ |
| 1 | ✣ | | ✣ | ✣ |
| 2 | | ✣ | ✣ | ✣ |
| 3 | | | ✣ | ✣ |
| 4 | ✣ | ✣ | | ✣ |
| 5 | ✣ | | | ✣ |
| 6 | | ✣ | | ✣ |
| 7 | | | | ✣ |
| 8 | ✣ | ✣ | ✣ | |
| 9 | ✣ | | ✣ | |
| 10 | | ✣ | ✣ | |
| 11 | | | ✣ | |
| 12 | ✣ | ✣ | | |
| 13 | ✣ | | | |
| 14 | | ✣ | | |
| 15 | no lines active | | | |

**✣ indicates respective line is at high (+5v) level with value shown in location 54020**

**Port 2 is controlled in the same way, using the formula (16*PORT2)+PORT1**

***Figure 1 - results of values POKEd into location 54020***

## ANYTHING HAPPENING?

Note that we cannot actually see the results of these POKEs unless we attach some sort of visual device (like an LED) between the appropriate +5v and 0v line being controlled. Tandy sell two low current, high efficiency red LED's for around 60p which you could play with.

Making connections to the fiddly pins of the controller jacks is not a good idea, of course, but we can use an old joystick cable for this

purpose. That old CX10 cable will do perfectly. Luckily (and helpfully) the six weedy cables buried within the thick black cable are coloured - a glance at Figure 3 will help you to determine which cable connects to which corresponding pin.

## TAKE NOTE

Here are a few things to note at this stage.

1. The ports can only handle a small load, so anything more than a simple LED will require the use of a reed relay to control a secondary, independent circuit. Don't expect the computer to supply enough current to run your Class A valve amplifier with electrostatic speakers!
2. Don't use lethal voltages in the secondary circuits. Make your robot run on batteries instead.
3. Be careful.

## WHAT USE IS IT?

What new uses does this new-found knowledge suggest? Well, you could investigate robotics (as already hinted), explore rudimentary data transfer between two or more computers using RAM in a functional second computer for slow access data storage, Local Area Networks (LAN), Control systems, printer interfacing (for technical junkies), Analogue to Digital Converters (ADCs) and even Atari controlled disco lights! I suspect that you already have your own ideas. ●

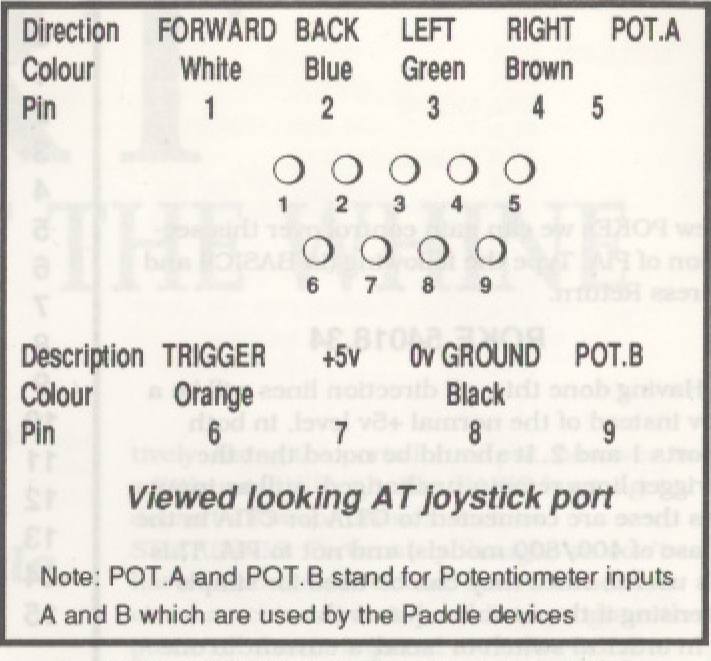



*Figure 2 - cable connections in cx10 joystick*

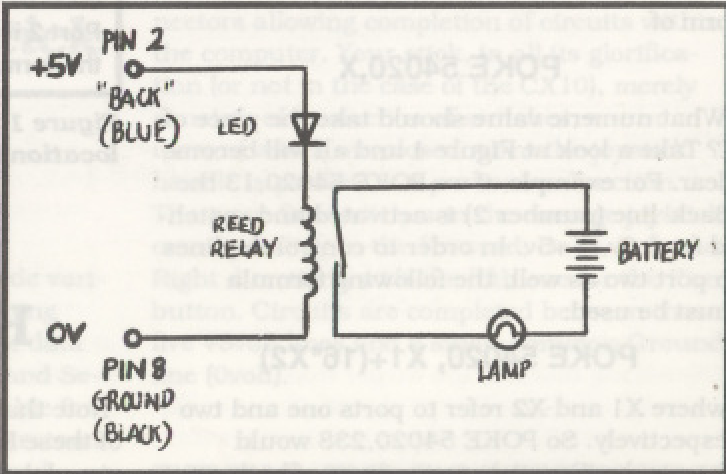



*Figure 3 - an example circuit*



# DISK BONUS

## MATCHBOX

### by Joel Goodwin

Matchbox is a game for two players where the object is to find pairs of shapes hidden behind an array of boxes. It is a test of memory (and perhaps a little luck).

After running the program in Basic, the program will require a period of time to initialise the graphics. Once this is complete the title screen will appear. You will need to press START to continue. The program will then ask you whether you have one or two joysticks. Respond to this by pressing '1' or '2' on the keyboard. You will then be asked to enter the name of each player (maximum 8 letters).

Now the game will begin. The player whose turn it is will find their name highlighted at the bottom of the screen, next to the player's score. That player will then use the joystick to move a cross around. The player must use this cross to select two green boxes which will reveal whatever shapes they are obscuring.

If the shapes are the same then the player scores one point and the boxes selected will take on the player's colour (blue or red). The player will then get another attempt to find a pair. If the shapes are different the player's turn is over. When the other player moves the joystick the shapes will be hidden again and then that player will have a go at trying to find a pair of shapes. The exception to this is the Joker shape - which is a big 'J'. This will pair up with any other shape, so if you reveal it then you have automatically earned one point.

The game continues until no more pairs of shapes can be found and the player with the highest number of points wins. You can play again with the same players if you wish but the player who went first last time will go second.

I hope you enjoy Matchbox - in my experience it can keep two players amused for many hours!

plus ...

## WORDGRID

### by David Preston

A version of the classic word puzzle game where a list of words is hidden in a jumble of letters. You have to find each word on the list using your joystick to highlight the first and last letters of each word. As a word is discovered it is highlighted in the word list and crossed through on the puzzle. The game becomes harder as more words are crossed through.

WORDGRID is written in Turbo Basic and is an excellent version of one of the classic games and full instructions for play are included in the program.

**These great programs are the BONUS on this issue's disk. If you are not a disk subscriber you can still obtain a copy for £2.95 from NEW ATARI USER, P.O. BOX 54, STAFFORD, ST16 1TB. Please make cheques payable to PAGE 6 PUBLISHING or order by telephone with your Visa or Access card on 01785 241153**

# HEY! HEY! it's The TIPSTER

*Well, The Tipster looks more healthy this issue than for many a month with several new tippers stepping forward with help and advice. It all makes for an interesting column once again. There are a few tips that have gone into a permanent limbo (or are trapped inside a dead hard disk drive) so if you have sent in anything to The Tipster in the last year or so which you haven't seen in the column, please send it in again and I guarantee to use it as soon as possible.*

## SOME MORE ANDY TIPS

Let's start off with the rest of Andy Guillaume's tips as promised last issue.

### RAMPAGE

There's no Time Limit during the game, so just wait at the side of the screen until the Snipers disappear and the Helicopters fly away. In a one player game ignore your opponent as he always dies anyway. Try to destroy the really small buildings as soon as possible as sometimes these can be hard to destroy if you don't smash the exact bit. Never grab Question marks as these are almost always 'Lose Energy'. Only grab Food or Points if the Helicopters have just swooped past.-

## THE CITADEL

Mr C Ayres from Wood Green in London points out that although we gave the last two codes for THE CITADEL in issues 72 and 73, he can't find the full codes in previous issues. As THE CITADEL featured as our disk bonus for Issue 74 he feels that you might like to know the full codes, so here they are.

| | | | | | |
|---|---|---|---|---|---|
| 1 | ----- | 10 | ZEBRA | 19 | SOLAR |
| 2 | SPACE | 11 | DREAD | 20 | TRITT |
| 3 | CLOUD | 12 | BLAST | 21 | TIGER |
| 4 | ALPHA | 13 | SWORD | 22 | VIDEO |
| 5 | KAPPA | 14 | CYBER | 23 | HYDRA |
| 6 | IMAGE | 15 | RINGS | 24 | EARTH |
| 7 | SUPER | 16 | PLUTO | 25 | TOUGH |
| 8 | PANIC | 17 | GENIE | 26 | ----- |
| 9 | MAGIC | 18 | STORM | | |

Mr Ayres also sent in the final message which you get when you complete the game but I feel that it would spoil things to see it in print. Better you find out for yourself, even if it means cheating on a few levels to get there.

### BANG BANK

Use the Keyboard controls (which are almost the same as the Spectrum version, West Bank). These are:

| | |
|---|---|
| Shoot at Left hand door | - 1 |
| Shoot at Middle door | - 2 |
| Shoot at Right hand door | - 3 |
| Move Left | < |
| Move Right | > |

### ATARTRIS II

On levels A-D, always try to get 4 lines at a time for maximum points. On level B take 1 line first before carrying on in 4's. On level C take 2, and level D take 3 lines. Try to get through level E with the build-up height as low as possible, and take all lines as they come. Then it's just a matter of getting to level Z with the height low enough to make it possible - it is!! My top score is 444865, 512 lines.-



# WHAT, NO PD TIPS?

*Although we have covered hundreds of commercial games since The Tipster was born, there has not been much help with Public Domain games so we welcome James Mathrick again with some help and advice on a couple of great disks from the Page 6 Library.*

## THE NEPHEW

This two disk set (Disk 101) comes from Germany which may be the cause of some difficulties. On my copy there were a few bugs with the loading of graphics, which meant that if I entered a certain room or situation, the computer would lock up. However, the game is written in Turbo Basic and is not protected at all so it can be easily listed to screen or printer. Firstly in order to wear something, use the verb 'carry' (the German 'trage' encompasses both verbs) and to drop something use the verb 'lay'. In order to leave the house, open the door with the keys and ensure you have shaved in the bathroom. Use the pole to open the trapdoor in the ceiling. Wear spectacles to get down to the cellar safely.

The bugs can easily be cured by finding them, then LISTing the program and correcting them. Make sure you have Turbo Basic loaded. *(Well, thanks James but what are the bugs??)*

## AURA ADVENTURES

This disk (251) claims that I am the only one playing adventures, but I hope not! Here are five excellent adventures on one disk with a brilliant menu screen that deserves anyone's attention. If you want to give it a try, here are a few tips that might help you on your way, combined with a few problems that I am having. Maybe someone can supply the answers to these next time.

### THE ISLAND

The aim is to escape the island you have been shipwrecked on, If you GO WATERFALL, you will find yourself in a secret passage and eventually in a cavern with a lantern in. I can't however light the lantern in order to light up the dark cave. You can use the large leaf as a float to swim the lake to an island, but what do you do with the brittle sticks? Am I supposed to break the glass bottle (X (examine) DEBRIS) into glass? There is a vine somewhere in the game which may be used for tying to the branch on the cliff face, or for getting up to the hole in the cavern roof.

### THE BOW OF BEURA

The aim of the game is to construct the legendary Bow of Beura. I have completed this one (LESCAT). *[No, I don't know what that means either! Tipster.]*

If you wander around you may come across a sad dragon. Maybe he would help you if he felt happy. Telling jokes won't cheer him up, so something else is needed to make him laugh. Maybe the eagle could help, but it may want something in return,

The bear is guarding a nice lute, but I doubt he prizes it highly seeing as he cannot play it. Maybe he needs something else to get his teeth into. As for the stroppy witch, maybe she's a tad stressed. She should listen to more music. The wizard's friendly enough but he needs some help himself. Feeling knowledgeable?

more ☞

The gargoyle's a bit stubborn though. You may need to be pushy. Feeling strong? You may find yourself in possession of seed. It may need to be planted somewhere nice and magical though.

Getting the magic branch may be tricky as a normal axe won't reach, a larger one may come in two parts. Xamining the scenery may provide the answer, although this problem may crack you up.

So you've got half of the bow, but what about the bowstring? Things get pretty hairy around now.

A good adventure this one with star appearances by the authors.

# THE BREATH OF LIFE PART 1

The Princess has fallen sick and you have been sent out to seek a legendary cure. Part 1, when completed, will give you a password to proceed to Part 2. I haven't reached that stage yet, but when I tried to load gamesave "D:B*.*" the game crashed and dumped the final message on the screen. Whoops!

I got a bit lost at the start being too occupied with the hole above the drawbridge. Ignore this, you will find what this is later. Concentrate on that locked door. You can't set fire to solid wood with the match provided but maybe it needs some help. Try Xamining the boiler. Where's that? Remember Zork? Wasn't there a mat and trapdoor puzzle there too?

Once you've got the door burning, you can't go through it. Maybe you could use the water provided but you may need that later. Did you try kicking the door before burning it? Serves you right. Maybe it's a bit weaker now it's burning though.

The sleeping, and quite obviously drunk, guard will react violently if you move, wake or attack him. He ought to recognise you, but by the look of the empty bottle on the control panel, he's had quite a lot of liquid already. Maybe you could fight fire with fire. But that's silly, you should fight it with water.

The bandits will get you if you're not careful. You have to fight them, but you need a vantage point. You may need the oil can to reach it though. Another container will be needed if you want to scare them off for good.

The empty jug looks pretty. Maybe someone in town likes jugs (in the literal sense). Maybe he'll swap it for gin. The old man said he could take you across the water if you paid.

Did you try going in the fountain? There's a grille in there, but to get it up you might need some rope. You might need the remnants of your attack on the bandits.

The owner of the house is not too friendly, so how do you get into his house without getting shot? God knows. This is where I get stuck. I give the gin to the old man, but he wants more and I can't get into the house. Help!

# CORSAIR!

You plan to go after legendary treasures belonging to an ex-pirate. You start in an inn, with a hefty bill and an empty purse. You will soon find a parchment and a lantern, and maybe some blankets. The most persistent adventurers will find a way through the window, but it's a long drop down. The barman won't let you through the door without paying, however, so it looks like it's the window. The inn is hardly class and the window ledge requires attention. The description of the town square will tell you of the local trade - maybe you could try your hand yourself! The results will not be much, but some may appreciate it more than others. The fat merchant's room may have rich pickings too, but his door is locked. Recycling twice may be the solution and the sea does not look too clean.

A slight glitch is that should you come across 100(?) silver doubloons, the beggar by the gate will not accept them - yet. He's more interested in small change. A swift sweep of the merchant's room will result in a couple of finds, but don't spend too long in there.

For fun, try shouting what is on the table in the pub.

Should you leave the mainland, you may find an island. A number of treasures need to be found and deposited in your cabin. One may need to be dug for, another's been overgrown over the years, and one went with the owner. Xamine the cliff face, check anything you find, and leave no trace but keep your respect for the deceased, because they're busy moping up in a tower. They're also guarding a pistol and a book. Not a very interesting book, but it could be worth a bit.

Someone got to the last treasure - he may need some friendly persuasion ... he who lives by the pistol. Luckily there's some shot outside.

This is where I become unstuck. I get all the treasures and deposit them all and finish the game, but I am still 10 points short. Has it anything to do with the fact I didn't kill the baddie, and merely shot him in the shoulder?

# ATLANTIS

Hmmm. I wonder what might happen to this place then? Might be an idea to get away. Should you pray in the temple, then you will receive more briefing. There are olives on the tree outside, and the mystic says the market is the place for making money, so off I go. Sell that, buy this. Off to the theatre. Only one actor left though, and he's not too happy. Cheer him up with a drink, so you can steal what's his. You aren't allowed in the marketplace temple, so you may need a disguise. Inside you might find something to light your candle with. And remember, although goddesses are immortal, they don't take too kindly to being thrown off a cliff. Maybe something here could help.

The mystic also says there are tunnels under the city, but I can't find them. And I can't drive away the Empusae either. And why is the palace deserted? And why does the mystic spew out cryptic characters now and again? Maybe only Alias Maximus and John E. could answer that.

Well, there you are. I hope that I have not raised more questions than I have answered, but if I have perhaps someone will give us some more answers next time. The Aura Adventures collection is an excellent selection, and there are other adventures in the making. Well, done Aura! Keep the tradition alive.

## ZORK

**We all want to find the fabled pot of gold, so Harry Lane has some advice on how to find it in ZORK. To go on the rainbow in ZORK I, go to the end of the rainbow and wave the sceptre at the rainbow. This reveals the pot of gold. To get out of the maze via the grating, Harry went W W W U GET KEY SW U NW D NE which is a different way from that given in Issue 75 and Harry says it's quicker.**

# HELP!!!

John Hull wants some info about **MERCENARY**, both games 1 and 2. In Mercenary 1, John has played through to the end but failed to find where two articles in the game go. These are the CHEESE and the COFFIN. John points out the the CHEESE can be used as a flight craft. John has not yet completed Mercenary 2 and begs for any clues and, possibly a map, to help him on his way. Finally John says he believes that there is a third Mercenary available on disk. Is that so?

Daniel Yelland supplied some tips this time and would like some help in return. In **RETURN TO EDEN**, Daniel wants to know how you go down the little island scene without dying. In **THE GOLDEN BATON**, how do you open the door and cool down the hot Quartz? Finally how do you cross the pit in **SPELLBOUND**?

Plenty there for you to help with so let's have some answers for the next issue.

***Thanks also to Miguel Letemplier, Jason Kendall and Daniel Yelland for sending in their tips which should appear next issue***

* * * * * * * *

# KEEP 'EM COMING

A nice lot of tips this time but that leaves the cupboard almost bare so keep the info coming. There are several requests for help raised in this issue so let's have some answers to those, as well as anything you might have discovered yourself on your favourite games. The address, as usual, is:

**THE TIPSTER**
**NEW ATARI USER**
**P.O. BOX 54**
**STAFFORD, ST16 1DR**




# TEXTPRO MACROS

***TextPRO is one of the finest Public Domain programs for the Atari Classic, probably second only to Turbo Basic and is the word processor most widely used today. In these articles Frank Walters explains how you can expand your use of TextPRO with its Macro facility***

In order to get the most out of this series of articles I first need to define a few conventions for the key presses you'll be using in connection with TextPRO macros. Keys on the keyboard will be enclosed by brackets. [START] means the START key. Inverse characters will be bracketed with 'less than' and 'greater than' symbols, e.g. <=> means inverse =, entered from the keyboard by first holding down [SELECT] then typing the [=] key. Multiple key strokes are indicated by an 'underline' symbol or _ connecting the indicated keys. [SELECT]_[CONTROL]_[V], means first hold down [SELECT], then hold down [CONTROL] and while holding down both keys press [V].

TextPRO versions 4.54 and earlier used the extension .MAC for macro files. Because of changes in the use of some macro command keys, the extension .MAX is used in TPX 4.55 and later. Older macro files may be converted to the new. The .MAX version macro system is discussed in this article.

## WHAT ARE MACROS?

Macros are simply designated keys that will type many keys for each macro key you type. If you have a file named TEXTPRO.MAX on your TextPRO disk, you are probably already using macros. The HELP files are an example of a simple macro. Your press [OPTION] and a number and the macro types the keys necessary to copy a particular file from the disk to the editor.

*Example:* You press [OPTION]_[1], the macro types:

[SELECT]_[CONTROL]_[Q] (copy command)
TPHELP.01,E: (filename, editor)
[Return] (execute copy command)

In this case the macro typed 14 keys while you only typed 2.

## HOW DO I USE OTHER MACROS?

First you have to load a macro, or set of macros, into the macro buffer. Macros are contained in disk files, which are actually text files created by typing the correct macro information in the TextPRO editor and saving it to disk. Macro files use the extension .MAX to identify them as such. There are three ways to load a macro file into the macro buffer:

1. Rename the macro file to TEXTPRO.MAX and it will load automatically from the disk
2. From TextPRO editor, use [CONTROL]_[V] and type the full macro name, then press [Return]
3. Macros can be loaded from other macros. You do this when you load a macro by pressing [START] and typing the macro name (without adding .MAX). With TEXTPRO.MAX in your macro buffer [START] calls a macro to load another macro. It pre-selects which macro key to automatically execute when the new macro is loaded; types [CONTROL]_[V]; and enters the input mode and adds .MAX to your filename before it types a [Return].

Once your macro file is loaded into the macro buffer, you have to know what macro keys are designated in that set of macros. You can look at the macro file in the editor. It might have some text information at the top. Many come with documentation files that explain their use. You could, of course, make your own macros.

## HOW CAN [START] BE A MACRO?

You trapped me. Two exceptions to the [OPTION] requirement are [START] and [HELP]. TextPRO has special code for those two keys which simulate two other macros:

[START] = [OPTION]_[#]
[HELP] = [OPTION]_[?]

In both cases, the normal keys are SHIFT keys ( [SHIFT]_[3] and [SHIFT]_[/] ). These macros are defined in the macro file by the # and ? symbols.

***TextPRO rule of thumb:***
[HELP], [START] and [OPTION] are only used to run macros. [SELECT] is used to type <inverse> keys.

# HOW CAN I MAKE MY OWN MACRO?

I'll give you one example here before going into further detail later in the articles. Clear the TextPRO editor with [SHIFT]_[Clear] and reply "Yes". We will program the simplest macro of all using the [START] key so we don't even have to hold [OPTION]. Remember, [#] is the key that [START] looks for, so that is the first thing we type into the editor to identify the macro key that will activate this macro. Press [SHIFT]_[3] to get the # on your screen.

Next we have to define it as a macro key with inverse =. Hold [SELECT] and press the [=] key. Everything entered after the <=> will be typed by the macro when you press [START]. Now type your message and press [Return]. Try something like this:

#<=>This is the [#] START key macro.

You might think that the [Return] you typed at the end of that line is the end of the macro. WRONG! Now type another line below that. Leave the 'NOT' or remove it, whichever you think is true.

I think this line will NOT print.

Save the macro to disk with [CONTROL]_[S] and type in the filename:

Save>D:DEMO.MAX

Now type [CONTROL]_[V] and press the space bar to enter the default filename:

Macro>D:DEMO.MAX

When you press [Return], you should see "No errors" in the prompt line, but nothing happened on the editor screen. That is for two reasons. First you only loaded the macro into the macro buffer with [CONTROL]_[V] and it will not execute automatically. Second, if you had used [START] to load the macro, you should just type DEMO but not .MAX since the START macro adds .MAX to whatever you type. Some DOSes might not like DEMO.MAX.MAX as a filename. Also you did not designate the pre-selected automatic macro key in your macro. # is for [START], but @ will auto-execute when loaded from the START macro (or in the TEXTPRO.MAX file when loading TextPRO from disk).

Press [START] several times and see what happens. You should see both lines that you had typed, even though the second line followed a [Return]. There are only three ways a macro terminates:

1. You define a macro following it with another inverse <=>
2. It tries to branch to an undefined macro
3. It runs out of keys in the macro buffer

If nothing happens when you press [START], go back and read this over and try again. You surely missed something and are not ready to go on from here.

# LET'S RECAP

Here's one last demo macro to review what we have discussed. Clear your editor; reload DEMO.MAX from your disk into the editor (not the macro buffer). Next, add these two additional macros to it:

#<=>This is the [#] START key macro.
I know this line will print.
@<=>This is the [@] Autorun macro.
?<=>This is the [?] HELP key macro.

Now, save this with the filename DEMO.MAX. First you have to re-load TEXTPRO.MAX into the macro buffer, since you can't use the START key to load another macro with the macro you have in memory right now! Press [CONTROL]_[V], and type "TEXTPRO.MAX" [Return] and you should get the "No Errors" message.

Press [START] and see "Macro>D:". Type DEMO and watch the ".MAX" being added to the name. See the @ "automatic macro" execute right away?

Run the demo using [HELP] (and [OPTION]_[SHIFT]_[/] ) and [START] (and [OPTION]_[SHIFT]_[3] ). Try the "automatic" macro again with [OPTION]_[SHIFT]_[8], which is the @ key.

Observe that each macro only types up to the beginning of the next macro. [START] prints two lines because the @ macro terminates the # macro at one character before the @.

Re-load TEXTPRO.MAX with [CONTROL]_[V] and play around with what you have learned before going on to the next lesson.

# INTERACTIVE MACROS

Let me first tell you a little story to explain why macros can be so useful. I was using a program the other day to create checksum bytes for BASIC line listings. I got a text file with checksums and a LISTED program with line numbers but had to figure a way to move the two-byte checksum from the one file and put it in front of the actual line in the LISTed file so that it would print out that way. I loaded the LISTed program into Bank 1 of TextPRO and the checksum file into Bank 2 and began to cut and paste, moving the checksum and space from Bank 2 to the appropriate line in Bank 1. It was working but it was WORK! After about six or seven moves I realised this was a job for a macro. I counted the keystrokes to move one checksum and return the cursor to where it started, but at the next line below. Including changing the checksums from normal video to inverse, it took 11 keys for each move.

I typed those 11 keys into a macro defined for the [START] key (#). I then tried it. Every time I pushed the [START] key it moved another checksum, changing it to inverse. With 100 lines, that reduced 1,100 keystrokes to 100. Not good enough. I added a "GOTO" at the end that looped back to the # key again, making a continuous loop once the START key was pressed. The macro contained 36 bytes and defined three other macros: [HELP] or [?] to reload TEXTPRO.MAX, (Autorun) or [@] to set ATASCII Mode for the inverse switch, and [START] or [#] to do the actual transfer.

I started over and this time one keystroke produced all 1,100 keys necessary to move the entire list of checksums from Bank 2 to Bank 1. Now that's efficiency! Of course I had to hit [Break] when it reached the end of the buffer and was no longer moving anything, but it sure made the job easier. It only took me three minutes to make the macro and now I have it available to do the job whenever it is needed. This is an example of a single task macro that can be loaded from disk whenever that task needs to be done, but is not needed for everyday use.

# THE MORAL

The point of the above is that you need to understand how to make your own macros to take advantage of the real power of TextPRO to make your job easier. Sometimes I make a macro for just one task and then don't bother to save it because it's not likely to be needed again. If, by chance, I need it again, it'll just take a minute or two to make another one.

# LET'S DO MORE

Earlier we discussed the theory of macros and how to load them into the macro buffer and execute them by holding down [OPTION]

and pressing another key or with special keys like [START] and [HELP]. We made a sample DEMO.MAX to show how these special keys work, along with the Autorun (@) macro key.

DEMO.MAX only types on the screen and is of little value other than instructional. Let's modify it to make it do some work for us. We'll start with the HELP key [?]. With the DEMO macro loaded into your editor, we are going to change the function of ?. Place the cursor on the 'T' in 'This' next to the inverse <=> and delete all after the equals sign. We are going to make the [HELP] key load the default TEXTPRO.MAX for us. The TextPRO command to load a macro is [CONTROL]_[V]. To type this in the editor you must hit the [ESC] key first so that the status line says "ESCape Set". Then type [CONTROL]_[V] and you will get the vertical line graphics character. This vertical line will be right against the inverse equals sign already on the screen.

What do we do after the [CONTROL]_[V] to load a macro? We type the filename. So do that. Type "TEXTPRO.MAX" and add a [Return] to that to execute the Load Macro command. The revised [HELP] macro should be entered as follows:

?<=>[ESC][CONTROL]_[V]TEXTPRO
.MAX[Return]

Now save the modified DEMO.MAX to disk and re-load TEXTPRO.MAX with [CONTROL]_[V]. Press [START] and type "DEMO" to load the new DEMO.MAX macro. After the Auto message press [START] and finally press [HELP] and see how it loads TEXTPRO.MAX back into the macro buffer. Repeat pressing [START], type "DEMO" and [Return] and see how we can now go easily back and forth between the two macro sets. This is what is meant by 'interactive macros'. You load a task-specific macro from disk with [START] and the filename, and you re-load the default TEXTPRO.MAX by pressing [HELP].

What if you have an Atari 800 and it has no HELP key? Good point. The question mark or ? is a SHIFT character, so an 800 can use [OPTION]_[SHIFT]_[/] which is 3 keys at the same time. To keep fingers from tripping over each other, 800 users can insert another macro for the [/] key:

/<=>[ESC],<[CONTROL]_[G]>?

A [Return] is not necessary following this macro. We are not executing a command like we were previously when loading that TEXTPRO.MAX macro. Again, [ESC] must be pressed to enter TextPRO's ESCape set mode before typing a CONTROL character. This time, [SELECT] must be pressed while typing [CONTROL]_[G] to get the INVERSE character. <[CONTROL]_[G]> is a special macro command key. All special macro command keys are <inverse> CONTROL characters. <[CONTROL]_[G]> means "GOTO" in TextPRO. It acts like the same command in Basic, but instead of a line number, it links to another macro. In this case it is telling TextPRO to GOTO the ? macro which is also the HELP key. This allows 800 users to press [OPTION]_[/] instead of [OPTION]_[SHIFT]_[/]. 800 owners may want to add this simple macro to all the macros that define the ? macro for the HELP key.

## MORE THAN ONE MACRO SET

You can fit a lot of macro commands into a 2k buffer and you have a lot of keys you can use but it becomes difficult to remember what all those macro keys are for. And some of the keys are a handful of fingers to press (e.g. [OPTION]_[SELECT]_[SHIFT]_[6] for inverse <&>). By defining different macros for specific tasks, you can call them from disk only when needed so as not to clutter the macro buffer. You can then return to your default macro set by linking with the [HELP] key like we did in DEMO.MAX. In fact, you may want to load TEXTPRO.MAX directly from other macros without using any key, once their job is done. Simply tag the "[ESC][CONTROL]_[V]TEXTPRO.MAX[Return]" command at the end of the last macro to be used.

Another advantage of "interactive" macro files is you can use the "easy" keys over again, such as [START] and the un-shifted keys and the pre-select Autorun feature.

## WRAPPING UP

We've seen how macros are able to load other macros, We also looked briefly at one of TextPRO's special macro command keys <[CONTROL]_[G]> which acts as a "GOTO". Next issue we'll cover all nine special macro keys in detail.

*This article originally appeared in the U.S. magazine Current Notes which, alas, no longer caters for the Atari Classic. We apologise to those readers who may have read the articles before but we believe that the information will be new to the majority of readers.*

### TEXTPRO UPDATE

**The versions of TextPRO currently in the PAGE 6 library (DS#8 and Disk 207) are earlier versions that use the .MAC macro extension. This issue we have added version 4.56, referred to in this article, as a 2-disk set with full documentation. Check out this issue's PD update for full details.**

# THE SOUND SELECTOR

**_Sound is a neglected area on the Atari yet good sound can help to enhance your programs no end. John Foskett here presents a superb program to help you - it will even write machine code routines for you!_**

The Sound Selector is a utility program designed to seek out all those weird and wonderful sounds available from BASIC. If the Atari can do it, the Sound Selector will find it. By careful control of the control register (location 53768), many different sounds can be created, from the roar of rocket engines to the sound of alien footsteps, from pops and whistles to the sound of heavy machinery. Once a particularly sound has been created, it can be saved to disk in various ways for inclusion into your own programs.

The sound data may be saved to disk as an assembler listing or in two forms of BASIC listings, firstly as SOUND statements with a POKE for the control register and secondly as a custom written USR loader routine for loading the sound data in the form of a string. There is a forth option available which is to save up to 100 sets of sound data into memory and then to save the entire memory to disk as a bank of DATA together with a program to action each set of data in turn.



## THE MAIN SCREEN

The main screen is a multi-coloured mode zero screen comprising of six sections each differently coloured and lined for clarity. Each section is described separately as below.

### *TOP SECTION (GREEN)*

This section is used to display the normal BASIC sound statements on the left hand side of the screen using a total of eight lines divided into four pairs by thin black lines to provide four sections, one for each channel. The sound statements appear on the upper of each pair of lines with the key which actions each parameter shown on the line below highlighted in inverse for clarity. On the right hand side of the screen are the two hardware control registers applicable to each channel, the frequency controlling register being on the upper line.

### *SECOND SECTION (RED/BROWN)*

This is a single line which contains the control register details underneath the hardware registers in the top section above such that they form a column of locations in descending order.

### *THIRD SECTION (BROWN)*

This section is devoted to the control register bits and comprises of a fully lined block of eight lines. This section contains a very brief description of the specific function of each of the control registers bits each preceded by the appropriate bit number from 0 to 7 and when set (that is 1), the numerals are highlighted in inverse. The total sum of the bits are displayed in the second section above.

### *FOURTH SECTION (BLUE)*

This is the information line and consisting of a single line, it is used to display the current mode of operation and any necessary comments.

### *FIFTH SECTION (PURPLE)*

This is a fully lined block of four lines used to display auxiliary information and when necessary, the save option menu. The auxiliary information displayed is the current session time in hours, minutes and seconds, the number of sound data sets currently in memory and the last used save file extension number together with the save option used.

### *SIXTH SECTION (RED)*

This is a single line at the bottom of the screen which is normally used to display "Press HELP for information", but is some cases it differs according to specific requirements.

## OPERATING DETAILS

When The Sound Selector is run, it counts and records all previously used sound file extension numbers found on the disk so that only unused extension numbers will be used for all save operations. In this way, all previously saved sound files are protected from being overwritten even if they are left unlocked. The Sound Selector will always use the lowest extension number available and will fill any gaps in the numerical sequence if any sound files have been deleted from the disk.

Note that The Sound Selector selects the sound file extension number itself automatically, so the number used should be noted for reference from the information line and from the auxiliary information displayed.

Because all sound files on the disk are counted and recorded, it is essential that the disk remains in the drive throughout the current session. If the disk is changed then The Sound Selector should be re-run to enable it to count and record the sound files (if any) on the new disk. If this is not done, then an error

will result during a save operation if the program attempts to save to a previously saved and locked file. If the file was unlocked, then it would be overwritten. However, after exiting from the error, the next save attempted will increment the sound file extension number and so the next save may be successful.

The Sound Selector is controlled entirely from the keyboard as follows.

# MANUAL CONTROL

Manual control is basically achieved using the keys "A to L" and the number keys "0 to 7". The letter keys are used to vary the parameters in the sound statements and the number keys are used to toggle the control register bits on and off. The letter keys are configured such that "A" controls the frequency of sound channel zero, "B" controls the distortion and "C" controls the volume. Sound channel 1 is controlled by the keys "D, E and F", sound channel 2 uses the keys "G, H and I" and channel 3 uses the keys "J, K and L".

The letter keys when pressed alone increment the parameters in the sound statements but when they are pressed with SELECT they decrement them. When the parameters have been incremented beyond their maximum, they are reset to zero and conversely, when they are decremented beyond zero, they are returned to their maximum value.

The frequency and volume parameters are incremented and decremented in steps of one, but the distortion parameters are incremented and decremented in steps of two because odd values used here only create a pop. The keys "A, D, G and J" may be pressed with START to enable a rapid increment of the frequency parameters, in which case, steps of 32 are used.

# AUTOMATIC CONTROL

***RANDOM NOISE:*** Pressing RETURN with SHIFT provides random noise where all the sound statement parameters are selected at random and all the bits of the control register are toggled randomly.

***SINGLE CHANNEL RANDOM NOISE:*** This is the same as above, but using only channel 0 and is operated by pressing the RETURN key alone. Only the frequency and distortion values are selected at random, the volume parameter is preset to 8.

***PURE TONES:*** Pressing RETURN with CONTROL provides random pure tones which is exactly the same as Random Noise above but with the distortion parameters preset to "10" to produce pure tones.

***MUSICAL CHORDS:*** Pressing RETURN with CONTROL and SHIFT together provides random musical chords which is the same as Pure Tones above but with the control register and therefore all its bits preset to zero.

***SOUND ON/OFF:*** Pressing the SPACEBAR toggles the sound on and off without effecting the current settings.

***RESET:*** Pressing CLEAR with CONTROL resets the program, returning all sound parameters and the control register to zero.

***PRESET:*** For convenience, there two preset facilities included. Pressing "P" with CONTROL presets all sound channels and pressing "O" with CONTROL presets only channels 0 and 1 leaving channels 2 and 3 zeroed. The frequency parameters are randomly selected, but the distortion and volume parameters are preset to 10 and 4 respectively.



***SAVE TO MEMORY:*** Pressing "M" with CONTROL saves the current sound data into memory, after which "Y" must be pressed to continue or any key to exit.

***CLEAR MEMORY:*** Pressing "X" with CONTROL clears the memory of all sound data, after which "Y" must be pressed to continue or any key to exit.

***SAVE TO DISK:*** Pressing "S" with CONTROL displays the save file menu which has four options available together with the option to press ESCAPE to exit. All files are saved to disk in the LIST format using the file name "SOUND.X" where "X" is a numerical extension number. The extension number used is selected automatically by the program using the lowest available and skipping over any previously used extension numbers. The four save options are as follows.

**SAVE OPTION 1:** Saves the current sound data as an assembler listing.

**SAVE OPTION 2:** Saves the current sound data as a BASIC routine containing the four SOUND statements and a POKE for the control register. The routine is written onto three lines with line 10 containing a REM statement with "SOUND.X" in inverse where "X" is the file name extension number. Line 20 contains the actual routine and line 30 forms a never ending loop (30 GOTO 30) to prevent the routine from ending so that the routine can be operated in direct mode. Line 30 must obviously be removed before the routine can be included in your own programs.

**SAVE OPTION 3:** Saves the current sound data as a custom written USR routine with the sound data stored in the form of a string. The routine is written onto 6 lines with line 10 containing the REM statement as before. Line 20 contains two DIM statements, the machine code string (MC$) and the sound data string (SND$). Line 30 defines the machine code string and line 40 defines the sound data string. Line 50 contains the actual USR command and line 60 forms a never ending loop similar to that previously mentioned above.

**SAVE OPTION 4:** Saves the contents of the entire memory to disk in the form of a bank of data together with a small program which loads and runs each set of sound data in sequence. Line 10 contains the same REM statement as before and the program itself is written from line 20 to line 60. A REM header for the sound data appears on line 70 and the data itself begins on line 80.

# INFORMATION SCREEN

The information screen may be accessed as prompted (normally at any time) by pressing HELP. The information screen gives brief details about operating The Sound Selector and OPTION is pressed to exit.

# THE DEMO SOUNDS PROGRAM

The accompanying demonstration program has been included to show off some of the sounds created with The Sound Selector. The data it contains was saved into The Sound Selectors memory and then saved to disk using save option 4 (see "Save option 4" above for more details). Note that this is the actual program that The Sound Selector writes, the only addition here is the REM header which has been added for clarity. The program is very simple so type it in and be amazed ....

```
SO 1 REM ***********************
AO 2 REM *   16 DEMO SOUNDS    *
SB 3 REM *    CREATED WITH     *
JB 4 REM * THE SOUND SELECTOR *
LY 5 REM *  BY  JOHN FOSKETT  *
LS 6 REM *    JANUARY 1996    *
SU 7 REM ***********************
FZ 10 REM Sound 1
YC 20 GRAPHICS 0:POKE 16,64:POKE 53774,64
   :POKE 752,1:? "[ESC,CLEAR][ESC,DOWN]WR
   ITTEN BY THE SOUND SELECTOR":? "
   [ESC,DOWN]PRESS START"
VC 30 RESTORE :? :FOR X=1 TO 16:? "SOUND.
   ";X:READ A,B,C,D,E,F,G,H,I,J,K,L,M
IL 40 SOUND 0,A,B,C:SOUND 1,D,E,F:SOUND 2
   ,G,H,I:SOUND 3,G,H,I:POKE 53768,M
UE 50 IF PEEK(53279)=6 THEN 50
LE 60 ON PEEK(53279)<>6 GOTO 60:NEXT X:GO
   TO 30
VK 70 REM Sound Data
SG 80 DATA 11,14,14,83,4,0,239,2,7,209,0,
   10,15
NM 90 DATA 255,10,8,254,10,8,0,0,0,0,0,0,
   8
RI 100 DATA 44,10,13,141,10,4,140,10,1,73
   ,10,11,158
UN 110 DATA 255,10,15,254,10,15,253,10,15
   ,252,10,15,0
KX 120 DATA 183,12,13,142,12,8,148,10,10,
   206,0,8,198
BH 130 DATA 194,14,3,28,12,12,175,8,9,82,
   10,14,188
CY 140 DATA 138,8,8,111,2,3,102,6,11,117,
   12,6,55
ZO 150 DATA 3,4,6,0,8,13,156,10,4,87,8,6,
   203
WB 160 DATA 207,10,9,163,10,9,84,10,11,20
   1,10,1,30
FF 170 DATA 165,10,10,219,10,4,72,10,12,2
   12,10,5,135
PH 180 DATA 143,10,8,183,10,3,226,10,1,15
   4,10,5,18
HL 190 DATA 255,10,9,97,10,15,197,10,1,33
   ,10,7,20
NY 200 DATA 101,4,12,77,4,0,226,12,14,75,
   12,5,242
DX 210 DATA 128,10,4,129,10,4,130,10,4,13
   1,10,4,0
EX 220 DATA 151,10,10,18,10,11,240,10,5,2
   15,10,9,250
UU 230 DATA 34,10,9,124,10,4,103,10,11,12
   8,10,7,152
```

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

# TECHNICAL DETAILS

There are several areas in the program listing which require some explanation as to how they work, these areas of the program as described as follows.

### THE INTERRUPT ROUTINES

The DLI routine actually contains six separate DLI routines to provide the programs multi-coloured screen and the first three are also used to provide the colours of the information screen.

The VBI is written as an immediate routine and defined on line 2260. The VBI is responsible for disabling the attract mode, disabling the CONTROL-1 toggle, to lock the keyboard into the uppercase mode and the reset the DLI vector address in order to synchronise the colours.

### PAGE 6

The DLI routine, the display list for the main screen and the display list for the information screen are loaded into Page 6 as absolute code. All three are defined as a string (U$) and MOVEd into page 6 on lines 2210, 2230 and 2250 at addresses 1536, 1614 and 1664 respectively.



### SCREEN FLIPPING

Flipping between the main screen and the information screen is achieved by switching between the two modified mode zero display lists. The main screen uses the normal screen RAM whilst the information screen uses a string (A$) dimensioned to 960 bytes for its screen RAM.

### PRINTING THE INFORMATION SCREEN

This is achieved on line 2290 where the screen RAM address is temporarily stored in the variable DP88 for later retrieval. The address of the information screen RAM (A$) is then DPOKEd into the vector (location 88) and the screen printed in the normal way with the data beginning at the end of the listing on line 2470 after which the vector is restored.

### SCREEN CLEARING

To display the SAVE menu and any error messages, it is necessary to clear a five line area at the bottom of the screen which is achieved on lines 1480, 1870 and 1970 using the MOVE command. This command MOVEs 200 clear bytes via Z$ into the relevant part of the screen RAM.

### DISPLAYING THE SESSION TIME

To enable the session time clock to run, it must be continually updated which is normally achieved via the operation loop on line 90. There are two other areas in the listing which also controls the clock which is in the two procedures MEMORY and CLRMEM via their continuous loop menu systems on lines 1200/1210 and 1300/1310 respectively.

### THE FILE NAME EXTENSION

During the programs initialising, it reads the disk's directory and records all the sound data files on the disk which have a numerical file name extension, all others files being ignored. All the previously used file name extension numbers are recorded in "F" array and the number of files counted is stored in the variable NUM which is achieved on lines 2410, 2420 and 2430.

When saving to disk, the contents of "F" array is compared with the variable EXT on lines 1590 and 1600 and if any match is found, then EXT is incremented by 1 and the comparison with the array repeated. When no match is found, the current value of EXT is used for the file name extension for the requested save and the array and the variable NUM is updated accordingly. A point to note is that EXT is also compared with zero (the unused elements in the array) and therefore EXT is always incremented to at least 1 which means that the first sound file saved to a new disk is always 1 and not zero.

# THE VARIABLES

One way to understand how a program works is to examine its variables.

### THE PROCEDURES

**ASSEM:** Writes the current sound data to disk as an assembler listing

**AUTO:** Initially resets and selects random values for the sound channel parameters and randomly toggles the control register bits

**AUX:** Prints the auxiliary information onto the screen

**BASIC:** Writes the current sound data to disk as a BASIC routine with SOUND statements and a POKE for the control register

**BEEP:** The beep

**BITS:** For toggling the various bits of the control register on and off, calculating the overall value and printing the result onto the screen

**CLEAR:** Used by RESET and AUTO to zero all parameters and to clear the display by printing zeroes in all positions

**CLRMEM:** Clears the memory of all sound data

**DIST:** For incrementing and decrementing the distortion parameters of all four sound channels by 2
**FREQ:** For incrementing and decrementing the frequency (period) parameters of all four sound channels by 1 (or 32)
**INFO:** Flips between the main screen and the information screen
**INIT:** Initialises the program
**MANUAL:** Determines if the word "MANUAL" is to be printed onto the screen
**MEMDATA:** Writes the entire memory to disk as a bank of DATA together with a simple program to load each set of sound data in turn
**MEMORY:** Stores sound data into memory
**PRESET:** Presets either all or just 2 of the sound channels
**PRINT:** Calculates the combined value of the distortion and volume parameters and prints the result onto the screen
**RESET:** Resets all sound parameters and the control register and prints zeroes in all positions
**SAVE:** Saves the current data or the entire memory to disk
**SND:** Used by the AUTO and PRESET procedures to print the parameters onto the screen
**TIME:** Prints the session time onto the screen
**TOGGLE:** For toggling the sound on/off and prints a message onto the screen accordingly
**USTRING:** Writes the current sound data to disk as a BASIC USR machine code routine with the sound data in the form of a string
**VOL:** For incrementing and decrementing the volume parameters of all four sound channels by 1

## LINE LABELS

**DECFREQ:** Begins the decrement section of the FREQ procedure
**DECDIST:** Begins the decrement section of the DIST procedure
**DECVOL:** Begins the decrement section of the VOL procedure

## STRINGS

**A$** The screen RAM string used for the information screen
**F$** Stores the file name used in the SAVE procedure
**H$** Defined as "PRESS <HELP> FOR INFORMATION"
**I$** Used to produce the inverse error number for the message in the error trap routine
**MEM$** The internal memory for storing sets of sound data
**OFF$** Defined as ": SOUND OFF"
**S$** String of 31 spaces used to clear the information line
**T$** Loaded from TIME$ to print the session time onto the screen
**U$** General purpose (U)tility string
**VBI$** The VBI routine
**Z$** String of zeroes used to clear an area of the screen (see Screen Clearing)

## ARRAYS

**A** Stores the three parameters of all four sound channels and the value for POKEing into the control register The array is configured as follows

| | |
|---|---|
| A(0) | Sound channel 0 Frequency |
| A(1) | Sound channel 0 Distortion |
| A(2) | Sound channel 0 Volume |
| A(3) | Sound channel 1 Frequency |
| A(4) | Sound channel 1 Distortion |
| A(5) | Sound channel Volume |
| A(6) | Sound channel 2 Frequency |
| A(7) | Sound channel 2 Distortion |
| A(8) | Sound channel 2 Volume |
| A(9) | Sound channel 3 Frequency |
| A(10) | Sound channel 3 Distortion |
| A(11) | Sound channel 3 Volume |
| A(12) | Control register |

**B** Stores the status of the bits of the control register as a 1 or a 0 where B(0) is the least significant bit. The array could therefore in effect be read as a binary number.
**F** Records all previously used sound file extension numbers found on the disk during initialising and is updated during each save operation.

## MAJOR VARIABLES

**DECREMENT:** Normally equals zero, but equals one if SELECT is pressed to enable the various parameters to be decremented in the FREQ, DIST and VOL procedures.
**DP88:** Represents DPEEK(88) and is used for storing the screen RAM address (See printing the information screen)
**DV:** Stores the combined value of the distortion and volume parameters for printing onto the screen
**EXT:** Stores the extension number of the last saved file name
**KEY:** Normally stores the internal code of the last key pressed
**MAN:** When equal to one it enables the word "MANUAL" to be printed onto the screen
**MEM:** Stores the number of sets of sound data currently in the memory (MEM$)
**NUM:** Stores the number of sound data files found on the disk during initialising and is updated as each save to disk is actioned
**OPT:** Stores the last used save OPTion
**PNT:** Points to the next position in the memory string (MEM$) where the next set of sound data will be stored
**P1 to P3:** Stores the colour values of the first three areas of the screen so that the colours can be restored when exiting from the information screen
**Z:** Toggled between zero and one and multiplied by the volume parameters of all four sound channels to turn the sound on and off

To conserve memory, many of the constants used in the program are replaced by variables prefixed by the letter "N" which makes their value readily identifiable, thus N4=4, N5=5, N6=6, etc. Note that these variables are used as if they were constants and that their values never change.

# AND FINALLY

Because of the timing requirements for the computer to run The Sound Selector program, the sounds created may not always sound the same when they are run via their saved routines. Even bearing this in mind, many fantastic sounds can still be created, so have fun and see what you can come up with. ●

# THE LISTING

***The SOUND SELECTOR program is written in Turbo Basic and is ready to run on this issue's disk. It is available as a TYPO coded type-in listing on request. Because of the difficulty involved with typing character strings, a special type-in listing is included to write the strings for you from a bank of numeric data. You do not need to use this program unless you wish. If you do use it, start by typing in this program then simply run the program and press START after which the strings will be written to disk using the filename "STRINGS.LST" in the LIST format. It is then a simple matter to ENTER them from disk.***

# The CLASSIC PD ZONE

*Our lifepod continues to drift with no sign of rescue. There is only one databank left to search. It must contain something we can use...*

## DRAWING POWER

If you own a 130XE and 1050 disk drive then you are lucky enough to be able to use DRAW7, a unique art package which is compatible with both Micro Illustrator and Micro Painter. Written by David Beifeld of Virginia, USA, Draw7 uses the extended memory of the 130XE to allow powerful operations not available with other 8-bit PD packages.

The package is shareware. By sending a small contribution to the author you will receive a disk full of files for use with Draw7.

A 16-page instruction manual and 3-page help file are included on the disk. They can be printed from the load menu. The help file can also be read from Draw7 itself. The manual is well written, easy to follow and will soon have you using the vast amount of features available.

You use the keyboard to access commands and a joystick (or the cursor keys) to move the cursor. All of the standard drawing commands are available including draw (graph mode), circle, ellipse, fill, line, rays, rectangle, plot, etc.

The Fill command is slightly different than usual in that the fill area is defined by two key presses which mark the top and bottom points. This is very useful for filling parts of an area but does slow down the operation when compared to the usual point and fill procedure. There are filled options to the circle, ellipse and rectangle commands to allow the creation of solid shapes. You can use any four colours from a palette of 256. Draw7 also lets you flip the colour registers around and change colours on-screen.

Dozens of other commands are included, e.g. Text, Undo, Zoom, Mirror, Flip, Invert, Load, Save, etc. The Text command allows you to place text anywhere on the screen. You can load a new font from disk and enter it on the screen (with wraparound) in any of six sizes. The Zoom command is actually a mode in itself which gives you the ability to load, save and manipulate the zoom window, e.g. replicate the zoom window in graph mode. The Home command takes the cursor instantly to the corners and centre of the screen. There are also Dotted Line and Wide Line options. There is a special screen movement function which gives you the power to move your picture up, down, left or right, each with screen wraparound (very handy!).

Draw7 features a powerful Mirror mode which operates differently to a standard mirror command. Rather than directly mirroring what you draw, Draw7 will duplicate and invert parts of the screen, e.g. horizontal screen half, vertical screen half and any quarter. This is obviously slower than direct mirroring but it is much more flexible in that you can duplicate and invert parts of the screen, as well as mirror. Other special effects include the ability to view the picture in Graphics 8 to 11, and flip the screen top to bottom or left to right.





There is a unique record/playback mode which lets you record the drawing of a picture and play it back at a later date as an animated sequence or to create unusual images over other pictures. A playback demo is included to show the results possible (press "6"). This record/playback mode is of little practical use but is very interesting nevertheless.

Overall, Draw7 is a very powerful package which brings a number of new commands to 8-bit computer art. Operation can be a little slow at times but the vast amount of commands available make it essential for any XE computer artist. The screen/picture manipulation commands are particularly impressive.

With BBK Artist, Pixel Artist Deluxe and Draw7 you can draw anything anywhere on the screen. Now all we need is a mega art package incorporating every feature of these three quality packages.

**CLASSIC PD ZONE RATING: 92%**

## SEE-YA!

*We were able to transmit a distress call by means of the playback facility in Draw7. It has been picked up by a science vessel studying the Classic PD Zone and we expect to be rescued soon. Surprised? No? Well I did consider a grisly finale to this story but opted for the good ol' happy ending. After all, if we all died horribly who would be here to do the reviews next time? See-ya in eight weeks.*

***The disk reviewed was:***

**DS#93 - DRAW7**

***(requires 130XE and 1050 density drive)***

**by Stuart Murray**



# UTILITIES

# MICRODOS

## by Andrew Pyrski

***If you have ever run out of room on a disk or just want to pack in as much as you can, try this super mini DOS which needs the minimum of disk space and the minimum of memory***

Microdos 1.3 is intended as a replacement for DOS 2.0 or 2.5 in situations were only the file read function is required in BASIC programs. If you have a number of programs that do not need to access DOS give it a try on your next disk.

## FEATURES

- DOS 2.0/2.5 compatible
- DOS.SYS and DUP.SYS are not required (save about 70+ sectors
- DOS 2.5 requires about 5.5K while MICRODOS 1.3 needs only 0.5K
- AUTORUN.SYS is not supported, but the basic file AUTORUN.BAS is automatically loaded and run if it is on the disk
- Read directory, write, format, delete etc. not accessible from BASIC
- BASIC OPEN (read only), ENTER, LOAD and RUN disk commands are supported
- The file spec MUST start with "D:" and MUST have a three character extension, e.g. "D1:MENU" must be rewritten as "D:MENU."
- BASIC DOS, SAVE and LIST disk commands are not supported

## USING MICRODOS

Type in the program and save a copy for future use. Make sure you have a disk formatted by DOS 2.0 or 2.5 and then RUN the program. This will write MICRODOS 1.3 onto the boot sectors of your disk. Copy Basic programs to your MICRODOS disk using a normal copy of DOS 2.5.

Note that MicroDOS will pick up and RUN any program on your disk named AUTORUN.BAS so you can add a menu program, or any other program that you wish to run automatically when you boot the disk.



```
EX 1 REM ################################
GK 2 REM #          MICRODOS            #
HB 3 REM #      by Andrew Pyrski        #
HB 4 REM # ---------------------------- #
DV 5 REM # NEW ATARI USER -MARCH '96   #
FC 6 REM ################################
NM 7 REM
CO 10 DISKIO=ADR("....")
BE 20 POKE DISKIO,104:POKE DISKIO+1,76:PO
   KE DISKIO+2,83:POKE DISKIO+3,228
EW 30 DIM BUFF$(128X3),I$(1)
MB 35 GRAPHICS 0
KK 40 ? "TO INSTALL MICRODOS 1.3":?
YO 50 ? "INSERT DISK FORMATTED BY"
SC 60 ? "DOS 2.0 (SINGLE DENSITY) OR"
BP 70 ? "DOS 2.5 (DUAL DENSITY ON A 1050
   DRIVE)";
FV 80 ? "AND PRESS RETURN":INPUT I$
ML 100 REM READ DATA
EA 110 FOR A=ADR(BUFF$) TO ADR(BUFF$)+383
NK 120 READ TMP:POKE A,TMP
CV 130 NEXT A
RV 140 REM WRITE SECTORS
OF 150 B=ADR(BUFF$):C=1:GOSUB 2000
RT 160 B=B+128:C=C+1:GOSUB 2000
RV 170 B=B+128:C=C+1:GOSUB 2000
GU 180 ? "MICRODOS 1.3 NOW INSTALLED"
TZ 190 STOP
NX 1000 DATA 0,3,0,7,21,7,76,3,8,214,7,25
   1,7,139,7,1
LB 1010 DATA 8,1,8,1,8,169,68,141,41,3,16
   9,9,141,42,3,169
RW 1020 DATA 7,141,43,3,169,0,141,3,8,96,
   160,1,162,11,169,32
SV 1030 DATA 202,157,7,8,208,250,200,177,
   36,201,46,208,4,162,8,208
CV 1040 DATA 245,157,7,8,232,224,11,208,2
   37,169,105,141,10,3,169,1
GZ 1050 DATA 141,11,3,169,8,133,43,32,191
   ,7,48,32,238,10,3,162
WK 1060 DATA 0,134,44,189,18,8,240,21,48,
   19,160,0,185,7,8,93
EM 1070 DATA 23,8,208,9,232,200,192,11,20
   8,242,164,44,96,165,44,24
CB 1080 DATA 105,16,170,16,220,198,43,208
   ,206,160,170,96,160,0,174,6
XI 1090 DATA 8,236,145,8,240,9,189,18,8,2
   38,6,8,160,0,96,140
JH 1100 DATA 6,8,160,136,173,144,8,141,10
   ,3,133,43,173,143,8,41
DL 1110 DATA 3,141,11,3,5,43,240,5,32,191
   ,7,16,207,152,96,160
UJ 1120 DATA 9,185,205,7,153,0,3,136,16,2
   47,76,89,228,49,1,82
DR 1130 DATA 64,18,8,7,0,128,0,173,3,8,24
   0,3,160,161,96,238
NG 1140 DATA 3,8,32,42,7,48,247,185,21,8,
   141,144,8,185,22,8
VY 1150 DATA 141,143,8,141,6,8,141,145,8,
   160,0,96,169,0,141,3
VP 1160 DATA 8,168,96,169,64,141,33,3,169
   ,8,141,34,3,169,0,141
MS 1170 DATA 4,8,169,21,133,12,169,7,133,
   13,169,148,141,231,2,169
QZ 1180 DATA 8,141,232,2,24,96,172,4,8,23
   8,4,8,185,76,8,201
PG 1190 DATA 155,208,10,162,0,142,33,3,16
   2,228,142,34,3,160,0,96
LY 1200 DATA 147,239,45,242,37,8,175,242,
   29,242,44,242,63,34,125,205
QM 1210 DATA 201,195,210,207,196,207,211,
   160,177,174,179,34,58,82,85,78
QE 1220 DATA 32,34,68,58,65,85,84,79,82,8
   5,78,46,66,65,83,155
NA 1230 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,
   0,0
FT 2000 REM DISK I/O
DY 2010 POKE 769,1:REM DRIVE 1
AZ 2020 POKE 770,87:REM WRITE WITH VERIFY
```

***continued on page 44***

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

## TURBO TYPE-IN

# 3-D GRAPH

***Mathematicians will be familiar with the graphs which can be plotted from different formulae which can be amazing to the uninitiated. Here Andy Guillaume presents a program and various formulas which can give stunning results***

The program plots a three-dimensional graph on a Graphics 8 screen from a chosen formula. It first calculates heights on the graph for X and Y positions then plots the resulting data. The program consists of a core calculation and plotting routine, and a Procedure in which the formula to be used is contained. Different formulae are stored as a LISTed file containing the following procedure.

```
1000 PROC FORMULA
1010 ...Calculation lines.
....
1030 S(N,I)=...
1040 ENDPROC
```

You must first LOAD the main program (3DGRAPH.BAS) then ENTER the required formula file.

```
LOAD "D:3DGRAPH.BAS"
ENTER "D:FORMULA1.LST"
RUN
```

Upon running, a countdown timer shows how many sets of coordinates are left to calculate before plotting begins. After plotting is finished press S to Save the screen to disk, remembering to change the filename for each subsequent picture. If you want to try another formula, DELETE 1000,9999 to clear the area and Enter another file as before. The Hidden Line effect is produced by purely plotting from the back of the screen to the front and deleting all data inside the current square, not perfect as some dots are usually missed, but try increasing Z on line 20 to draw more lines inside each square. Also try reducing the X Step, XS, to give a smaller width, and the Y step, YS, which tilts the graph towards or away from you. SC controls the height scale factor in the formula and should be adjusted so that the data does not go off the screen, causing a 'cursor out of range' error, for which the program does not check.

As is, the program plots Black squares and White surround. Swap the colour values on lines 90+100 for an inverted colour graph. You can turn off the Hidden Line attempt by setting FL on line 90 to 0.

To write your own formula, first Enter a previous one in the normal way then alter the terms to suit. Remember to keep a *SC to scale the results.

In some of the supplied formulas, R is calculated and gives the Radius from the centre of the grid. The grid for coordinates is a 30,30 array giving 900 points. When you have finished, Renumber the formula, i.e.

```
EX 1 REM ################################
ES 2 REM #          3D GRAPH            #
SG 3 REM #     by Andy Guillaume        #
HB 4 REM # ---------------------------- #
TN 5 REM # NEW ATARI USER - MAR 1996    #
FC 6 REM ################################
NM 7 REM
WI 20 XS=4:YS=%2:SX=159:SY=95-(YS*30):SC=
   8:Z=10:DEG :DIM S(30,30)
GA 30 GRAPHICS %0:POKE 752,%1
HM 40 FOR N=%1 TO 30:POSITION %0,%0:? 30-
   N;" ":FOR I=%1 TO 30
SO 50 EXEC FORMULA:S(N,I)=INT(S(N,I))
RQ 60 NEXT I:NEXT N
ZF 70 GRAPHICS 24:SETCOLOR %1,%0,12:SETCO
   LOR %2,%0,%0
PN 80 FOR N=%1 TO 29:X=SX:Y=SY:FOR I=30 T
   O %2 STEP -%1
JH 90 FL=%1:COLOR %0:EXEC DSQ
UM 100 FL=%0:COLOR %1:EXEC DSQ
IY 110 X=X-XS:Y=Y+YS
LX 120 NEXT I:SX=SX+XS:SY=SY+YS:NEXT N
VU 130 GET A:IF A=83 THEN OPEN #%1,8,%0,"
   D:FILENAME.PIC":BPUT #%1,DPEEK(88),768
   0:CLOSE #%1
MV 140 GOTO 130
UJ 150 PROC DSQ
FA 160 X1=X:Y1=Y-S(N,I)
VT 170 X2=X+XS:Y2=Y+YS-S(N+%1,I)
EH 180 X3=X:Y3=Y+(YS*%2)-S(N+%1,I-%1)
WO 190 X4=X-XS:Y4=Y+YS-S(N,I-%1)
IA 200 IF FL=%0
TX 210 PLOT X1,Y1:DRAWTO X2,Y2:DRAWTO X3,
   Y3:DRAWTO X4,Y4:DRAWTO X1,Y1
MQ 220 ELSE
MQ 230 DX1=(X1-X4)/Z:DX2=(X2-X3)/Z:DY1=(Y
   4-Y1)/Z:DY2=(Y3-Y2)/Z
QR 240 FOR Q=%0 TO Z
RP 250 PLOT X1,Y1:DRAWTO X2,Y2
VU 260 X1=X1-DX1:X2=X2-DX2:Y1=Y1+DY1:Y2=Y
   2+DY2
JI 270 NEXT Q
IZ 280 ENDIF
WC 290 ENDPROC
VW 1000 PROC FORMULA
JE 1010 R=SQR((ABS(16-I)*ABS(16-I))+(ABS(
   16-N)*ABS(16-N)))
QV 1020 S(N,I)=(COS(R*60))*SC
YL 1030 ENDPROC
```

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

*Above - the main program. Overleaf - six different formulae for you to try out*

```
VW 1000 PROC FORMULA
XK 1010 S(N,I)=(COS(I*20)-SIN(N*20))*SC
YI 1020 ENDPROC

VW 1000 PROC FORMULA
JE 1010 R=SQR((ABS(16-I)*ABS(16-I))+(ABS(
   16-N)*ABS(16-N))
BQ 1020 S(N,I)=(COS(R*60)+(2*COS(R*30)))*
   SC
YL 1030 ENDPROC

VW 1000 PROC FORMULA
JE 1010 R=SQR((ABS(16-I)*ABS(16-I))+(ABS(
   16-N)*ABS(16-N))
FZ 1020 S(N,I)=(SIN(R*40)+(2*COS(R*60)))*
   SC
YL 1030 ENDPROC

VW 1000 PROC FORMULA
JE 1010 R=SQR((ABS(16-I)*ABS(16-I))+(ABS(
   16-N)*ABS(16-N))
QV 1020 S(N,I)=(COS(R*60))*SC
YL 1030 ENDPROC

VW 1000 PROC FORMULA
JE 1010 R=SQR((ABS(16-I)*ABS(16-I))+(ABS(
   16-N)*ABS(16-N))
SX 1020 S(N,I)=(SIN(R*60))*SC
YL 1030 ENDPROC

VW 1000 PROC FORMULA
JE 1010 R=SQR((ABS(16-I)*ABS(16-I))+(ABS(
   16-N)*ABS(16-N))
IP 1020 S(N,I)=(SIN(R*60)+COS(R*30))*SC
YL 1030 ENDPROC
```

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

**RENUM 1000,1000,10**

and then List it out i.e.

**LIST "D:FORMULA9.LST",1000,9999**

## LINE COMMENT/USAGE

| | |
|---|---|
| 20 | Setup variables |
| 40-60 | Setup array |
| ..50 | EXEC FORMULA |
| 80-120 | Plot graph |
| ..90 | Fill on, C.0, plot square |
| ..100 | Fill off, C.1, plot square |
| 130 | Wait for Keypress, if S, Save screen |
| 150-290 | Plot and fill square routine |
| 1000- | Formula procedure goes here |

## MICRODOS

**continued from page 41**

```
ZZ 2030 POKE 773,INT(B/256):REM MSB ADDRE
   SS
LW 2040 POKE 772,B-PEEK(773)*256:REM LSB
   ADDRESS
TS 2050 POKE 779,INT(C/256):REM MSB SECTO
   R
AP 2060 POKE 778,C-PEEK(779)*256:REM LSB
   SECTOR
WK 2070 POKE 725,128:POKE 726,0:REM SECTO
   R SIZE
JU 2080 TMP=USR(DISKIO)
BB 2090 RETURN
```





UTILITY

# DISCNOTE

***If you can't remember exactly what the programs on your disks are, try H.S. Wood's utility which lets you add information to a printout of the disk directory***

When a disk directory is listed to a printer it usually only shows the filenames. This program allows notes to be put at the side of the filename to remind you what the program does or how to use it. As an example you might have the following entry:

STRANGE.CMD M/CODE run with X=USR(34816)

In addition notes can be put at the top of the printout as a reminder of the most important routines on a particular disk. I keep a printout for each disk in the disk jacket.

## PRINTER OUTPUT

The first part of the program (lines 10 to 40) set up the printer and ask for a date to be entered. Lines 15 16 and 17 set up the printer but I have left them as REM's because different printers need different codes. The program will work without these lines but if printer codes are used you should cancel them at the end of the program. (See REM line 30260)

A copy of the program should be put on each disk so that it can be loaded and run whenever a directory printout is needed. Line 40 should be changed to suit the particular disk e.g.

40 LPRINT "Special disk (front) dated ";C$

The user enters lines 50 to 999 as required to suit the particular disk. These lines, if used, will print out special notes for the disk and should take the following format:

50 LPRINT "This disk has a special DOS on it"

If no lines are put here then only the disk name and date will be put before the directory listing.

continued ☛

```
EX 1 REM ##################################
AR 2 REM #           DISCNOTE             #
SJ 3 REM #        by H.S. Wood            #
HB 4 REM # ------------------------------ #
DV 5 REM # NEW ATARI USER -MARCH '96      #
FC 6 REM ##################################
NM 7 REM
TC 10 ? "[ESC,CLEAR]":DIM C$(20)
SI 15 REM LPRINT CHR$(27);CHR$(108);CHR$(
   3);:LPRINT CHR$(27);CHR$(15);
TV 16 REM LPRINT CHR$(27);CHR$(83);CHR$(1
   );
VC 17 REM LPRINT CHR$(27);CHR$(51);CHR$(1
   6);
YV 20 ? "ENTER DATE"
YW 30 INPUT C$
BF 40 LPRINT "  NEW ATARI USER 1 (FRONT)
   DATED ";C$;
GH 45 REM ***  THE NEXT LINES WILL  ***
IS 46 REM ***  PRINT AT THE TOP OF  ***
PY 47 REM *** THE DIRECTORY LISTING ***
FK 50 LPRINT " (DOS 2.5) ":LPRINT
RV 60 LPRINT " DISCNOTE.BAS IS A BASIC PR
   OGRAM"
SC 65 LPRINT "WHICH WILL PRINT NOTES AT T
   HE SIDE"
LN 70 LPRINT "OF EACH FILENAME IN THE DIR
   ECTORY"
MD 75 LPRINT " THE NOTES MUST BE IN THE D
   ATA LINES"
BB 80 LPRINT "STARTING AT LINE 1000."
IB 999 REM *** DATA LINES START HERE ***
VU 1000 DATA DOSSYS, STANDARD DOS 2.5
XE 1010 DATA DUPSYS, STANDARD DOS 2.5
UF 1020 DATA DISCNOTEBAS, SEE NOTE ABOVE
ZC 1030 DATA LETTERCOR, USE PAGE 6 WRITE
   TO PRINT
FG 1040 DATA WRITEUPCOR, USE PAGE 6 WRITE
    TO PRINT
AR 30000 DIM A$(100),B$(80):P=0
GD 30005 RESTORE 1000:LPRINT
AI 30010 ?
NV 30020 OPEN #3,6,9,"D:*.*"
YI 30030 TRAP 30240
MB 30040 INPUT #3,A$
FJ 30050 LPRINT A$;
QY 30060 A$=A$(3,LEN(A$)-4)
LN 30070 A=1
RG 30080 IF ASC(A$(LEN(A$),LEN(A$)))=32 T
   HEN A$=A$(1,LEN(A$)-1):GOTO 30080
DF 30090 K=ASC(A$(A,A)):IF K=32 THEN 3011
   5
OL 30100 IF A=LEN(A$) THEN 30130
CH 30110 A=A+1:GOTO 30090
NK 30115 FOR J=A TO LEN(A$)-1:A$(J,J)=A$(
   J+1,J+1):NEXT J:A$=A$(1,LEN(A$)-1)
AS 30120 GOTO 30070
VZ 30130 B$=A$
QR 30140 A$=B$:TRAP 30220
VW 30150 RESTORE 1000
IF 30160 READ B$
DW 30170 IF B$=A$ THEN 30190
BP 30180 GOTO 30160
IR 30190 READ B$
GF 30200 IF LEN(B$)=0 THEN 30220
OV 30210 LPRINT B$:GOTO 30230
BB 30220 LPRINT
QH 30230 TRAP 30240:GOTO 30040
CB 30240 LPRINT :LPRINT :LPRINT :LPRINT :
   LPRINT :LPRINT
NX 30250 ? "[ESC,CLEAR] LOAD";CHR$(34);"D
   1";CHR$(58);"DISCNOTE.BAS";CHR$(34):LP
   RINT CHR$(27);CHR$(84);
RO 30260 REM LPRINT CHR$(27);CHR$(50);:EN
   D
YR 30261 END
```

Underline = INVERSE CHARACTERS - [ ] = CONTROL + CHARACTER - < > = INVERSE CONTROL + CHARACTER





# ATARI SAVED!

### *A special report from Les Ellingham*

Atari have officially announced that they have developed the ultimate in computer technology, a product that will ensure Atari's complete domination of the global computer market. The ATM is a quantum leap above current 64-bit technology, in fact an incredible 64-million bit machine that can process trillions of instruction per second. It is rumoured to be even more advanced that the Transporter system on the Starship Enterprise.

The ATM is the Atari Time Machine, a device that can transport its user back in time to any selected point. Atari plan to build a number of these devices and have a selected member of the Tramiel family use one to go back to the late 1970's. He will carry with him the marketing plans of Apple, IBM and a number of other successful computer companies and apply them to the best computer around at that time. History will be changed and instead of the world being dominated by computers that need a dozen different expansions to make them run indifferent programs, everyone will be running Atari machines that will be able to run top class applications straight from the box.

At least that was the plan. Rumour has it that the machine is now in the hands of the current marketing department of Atari who can't figure out just how to promote it. They are hoping that enough people will find out about it by word of mouth so that they can start taking orders. No delivery date can be promised at the present time, however, and the suspicion is that the first unit will not be ready until after the company has gone bust. The price is expected to be slightly more than anyone is willing to pay.

# DISCNOTE continued

## HOW IT WORKS

When the program has printed out the contents of lines 10 to 999 it goes to line 30000 which opens the disk directory. As each item of the directory is read the program searches the DATA lines to see if there is a matching filename. If there is a match then the filename is printed, followed by the remarks in the DATA line. Otherwise only the filename is printed. After each filename is printed the program loops until the directory is complete.

The DATA lines at 1000 must have the filename of the file which needs a note. The filename must be complete and must end with a comma. Full stops must not be included i.e. DISCNOTE.BAS must be DISCNOTEBAS.

The note must follow the comma, for example:

1000 DATA DISCNOTEBAS, Basic program to print a directory

Finally the program ends by printing LOAD"D1:DISCNOTE.BAS" to the screen. This is so that if several disks are being dealt with it is only necessary to move the cursor over the message and press return to load the file from the next disk.



# TUTORIAL

# THE XIO FILLS

***Ann O'Driscoll was pipped at the post last issue by our article on XIO but she has managed to salvage some of her proposed article with a useful demonstration of the FILL routines***

The program shown here is a follow up to one aspect of Paul Hollins' article in Issue 75 on the Atari's XIO functions. The listing gives several examples of XIO 18 - the FILL command - in operation, showing different features of its use in practice.

As Paul pointed out, the Atari fills over background colour by working one line at a time from left to right, the left edge of the shape being defined by an imaginary line going from a PLOT, or your last DRAWTO statement at the top, to a point set by a POSITION statement at the bottom.

The first few demos in the listing show different attempts to fill or partially fill an orange triangle drawn on a Graphics 7 screen. The triangle is set up by the subroutine at LINE 500 and has co-ordinates at 50,0 (top), 100,50 (bottom right) and 0,50 (bottom left). The REMs should give a fair idea of what's going on. In the first example, LINES 48 - 50 make the 'imaginary left line' equal to the triangle's left edge, so the whole shape gets filled in green, the colour selected in LINE 53. In example 2, an orange line is drawn in the middle of the triangle (LINE 60) before the XIO routine is called up. This acts as a new right edge. Filling starts as in example 1, but when the computer meets a non background colour it drops down a line before beginning to fill from left to right again. In example 3, the POSITION statement in LINE 85 makes the Atari stop colouring before it gets to the end of the triangle, while in example 4 the PLOT co-ordinates in LINE 100 direct the computer to begin filling below the top of the pre-defined shape. Examples 5 and 6 simply use PLOT and POSITION to define left edges which are different to the triangle's (LINES 115 and 140), so XIO creates a new left bor-



```
EX 1 REM ################################
BJ 2 REM #        XIOFILL DEMO          #
GE 3 REM #             by               #
WX 4 REM #       ANN O'DRISCOLL         #
HC 5 REM # ---------------------------- #
EP 6 REM #  NEW ATARI USER - MAR '96    #
FD 7 REM ################################
NN 8 REM
JD 30 REM __ EXAMPLE 1
BE 32 REM
PU 40 GOSUB 500
NS 45 ? "XIOFILL DEMO: Colour in this sha
  pe":GOSUB 610
XE 48 PLOT 50,0:REM start at top left
FI 50 POSITION 0,50:REM stop at end left
ZG 53 POKE 765,2:REM colour register
KW 55 GOSUB 600:GOSUB 610
KR 59 REM __ EXAMPLE 2
IB 60 GOSUB 500:PLOT 50,20:DRAWTO 50,50
RJ 65 ? "New line will act as a right edg
  e":GOSUB 610
YT 70 PLOT 50,0:POSITION 0,50:POKE 765,2:
  REM plot top left,position end left,se
  lect colour
KY 75 GOSUB 600:GOSUB 610
LN 79 REM __ EXAMPLE 3
ZK 80 GOSUB 500:? "You needn't fill to th
  e end.. "
DL 85 PLOT 50,0:POSITION 25,25:POKE 765,2
  :REM plot top left,position end left,s
  elect colour
KQ 90 GOSUB 600:GOSUB 610
LZ 94 REM __ EXAMPLE 4
XP 95 GOSUB 500:? "... or start at the to
  p.."
SM 100 PLOT 25,25:POSITION 0,50:POKE 765,
  2:REM plot top left,position end left,
  select colour
LM 105 GOSUB 600:GOSUB 610
QD 109 REM __ EXAMPLE 5
OX 110 GOSUB 500:? "... or fill the whole
  shape."
JX 115 PLOT 50,0:POSITION 50,50:POKE 765,
  2:REM plot top left,position end left,
  select colour
LB 120 GOSUB 600:GOSUB 610
RC 129 REM __ EXAMPLE 6
TU 130 GOSUB 500:? "You may start IN the
  shape too.."
OQ 140 PLOT 50,10:POSITION 0,50:POKE 765,
  2:REM plot top left,position end left,
  select colour
LU 145 GOSUB 600:GOSUB 610
SB 149 REM __ EXAMPLE 7
QA 150 COLOR 1:PLOT 140,25:DRAWTO 115,0:R
  EM end right to top left
KT 155 ? "So use XIO to define the left e
  dge..":GOSUB 610
OC 160 POSITION 90,25:POKE 765,2:REM stop
  at end left;pick colour
LY 165 GOSUB 600:GOSUB 610
TA 169 REM __ EXAMPLE 8
UP 170 GOSUB 500:? "The end left is below
  the end right!"
JF 175 PLOT 50,0:POSITION 0,60:POKE 765,2
  :REM plot top left,position end left,s
  elect colour
MC 185 GOSUB 600:GOSUB 610
TZ 189 REM __ EXAMPLE 9
LA 190 GRAPHICS 7:COLOR 1:PLOT 50,70:DRAW
  TO 100,50:DRAWTO 80,30:DRAWTO 80,20:DR
  AWTO 50,0:REM end right to top left
IA 200 POKE 752,1:? "A jagged right edge
  is easy.."
DG 210 POSITION 0,70:POKE 765,3:REM end l
  eft and colour
LC 220 GOSUB 600:GOSUB 610
BW 249 REM __ EXAMPLE 10
```

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

```
EE 250 PLOT 120,70:DRAWTO 120,0:DRAWTO 50
   ,0:REM end right to top left
IA 260 ? "..but a jagged left is done in
   steps"
SP 265 POSITION 80,20:POKE 765,2:GOSUB 60
   0:GOSUB 610
LY 270 PLOT 80,20:POSITION 80,30:POKE 765
   ,1:GOSUB 600:GOSUB 610
HX 280 PLOT 80,30:POSITION 100,50:POKE 76
   5,2:GOSUB 600:GOSUB 610
UF 290 PLOT 100,50:POSITION 50,70:POKE 76
   5,1:GOSUB 600:GOSUB 610
YG 400 ? "[ESC,DOWN]That's all there's to
    it - press a   key for Graphics 0":P
   OKE 764,255
BS 410 IF PEEK(764)=255 THEN 410
BM 420 POKE 764,255:POKE 752,0:GRAPHICS 0
   :END
OW 462 POKE 765,C:XIO 18,#6,0,0,"S:":NEXT
    C
JF 499 REM  - DRAW TRIANGLE IN GR.7
PA 500 GRAPHICS 7:POKE 752,1:COLOR 1:PLOT
    50,0:DRAWTO 100,50:DRAWTO 0,50:DRAWTO
    50,0:RETURN
OY 599 REM  - XIOFILL
MY 600 XIO 18,#6,0,0,"S:":RETURN
YQ 609 REM  - KEYPRESS ROUTINE
IQ 610 POKE 764,255:POKE 752,1
HP 615 ? "[ESC,DOWN][ESC,TAB]Press a key
   to continue"
FA 620 IF PEEK(764)=255 THEN 620
XJ 630 POKE 764,255:? CHR$(125);:RETURN
```

Underline = INVERSE CHARACTERS · [ ] = CONTROL + CHARACTER · < > = INVERSE CONTROL + CHARACTER

der as it colours. Example 7 shows how this technique can be used as a short-cut for drawing and filling objects: LINE 150 just draws the right edge of the shape, while XIO takes care of the left edge and the bottom boundary as it fills.

We saw from example 3 that not all of the object gets filled when the lower left (as defined by POSITION) is above the lower right of the shape. Example 8 shows what happens when the lower left is below the lower right - LINE 175 uses POSITION to place the end of the imaginary line lower down on the screen than the right hand side of the triangle. Because of this, the fill colour spills out over the rest of the screen as there's no right boundary to contain it.

The last two demos show how XIO FILL copes with unevenly shaped sides. As you might guess from the way the command works, the computer has no trouble with colouring shapes which have jagged right edges. It just carries on filling, line by line, until it meets a right border, then it drops down to the next line and starts again, working from left to right. Example 9 draws a crooked right-edge on the screen (LINE 190) to show this in operation. Shapes with uneven left edges, on the other hand, can be a bit tricky, because the left margin of our filled shape (i.e. our imaginary line between top and bottom) must be straight. The solution is to split the shape up into smaller sections, each with a straight left edge, and fill each one in turn. Example 10 does this. The XIO routine is called up 4 times (LINES 265 - 290). The lower left edge of the first segment (defined by POSITION in LINE 265) becomes the starting point for the second block (defined by PLOT in LINE 270): the end of the second block (POSITION in LINE 270) becomes the starting point for the third block (PLOT in LINE 280), and so on. The routines also vary the fill colour and introduce a pause routine after each segment is filled, so that you can see clearly what is going on.

Finally, some of you who read Paul's comprehensive article in Issue 75 on the XIO commands might like to check out a program called Tinydos on Page 6's very first utility disk in the public domain library (PD Disk #3). This uses XIO to rename, delete and lock/unlock disk files, all from within a Basic program. ●





# FREE NETS

***If the cost of on-line services worries you, the future may hold some good news as Gordon Hooper explains***

John S Davison expressed an interest in the concept of Free-Nets during an e-mail discussion we had after I contacted him regarding his article in the Nov/Dec '95 issue of NAU. It really is marvellous that we can talk between England and Canada so quickly and at so low (or no if you're on a Free-Net) cost. I normally received his reply either the same day or next day. And yes, you can get on the Internet with an 8-bit, with minor limitations. The main one is lack of graphics when you browse the World Wide Web (WWW).

For the 8-bit you will need a terminal program (I recommend BobTerm) and and Internet Service Provider (John uses Compuserve, I use a Free-Net). By using BobTerm, you can also use an XEP80, which makes reading a lot easier, as the Internet uses 80 columns.

A Free-Net is a Bulletin Board System (BBS) which furnishes information provided by community organisations, individuals, business and government at no (or minimal) cost to the user. It also provides Internet access to e-mail, newsgroups (such as comp.sys.atari.8-bit and comp.sys.atari.st, amongst literally thousands of others) and the World Wide Web (WWW). Also provided are cybercasting services from around the world. Cybercasting is defined as the distribution of electronic information to affiliate computer systems from a central computer. The Free-Net also provides local and international library catalogues, weather services and the NASA Spacelink.

## IN THE BEGINNING

The first Free-Net was opened in Cleveland, Ohio, in 1986 by Dr. Tom Grundner. He has Masters Degrees in human learning and education and a Doctorate in education philosophy. He is also very active with microcomputers and telecommunications and envisioned a BBS patterned after PBS in the USA. PBS provides educational and cultural TV broadcasting with financial support from governments, business and the public.

The Free-Net system also takes inspiration from public libraries. When the price of books went down and a rise of literacy occurred in

the 1800's, free libraries became established as a means of teaching and informing the public and are common today. Doctor Grundner wants to do the same thing through microcomputers now that they are within the reach of ordinary people. The concept of a free community computer information service is now here. There are Free-Nets in existence around the world as of this writing.

## LOCAL CONNECTION

The Free-Net I use is the Victoria Free-Net Association (VIFA) in Victoria, British Columbia, Canada. It was among the first, being the 12th one set up. It was started by a retired couple, Gareth and Mae Shearman who had educational backgrounds and an interest in telecommunications. They felt it would be of immense use to schoolchildren and of great interest to any person who had a microcomputer and a modem. This was just before the hype about the 'Information Superhighway' became common, and VIFA soon became a free onramp onto it.

The Shearmans started by forming a government-registered non-profit society to handle the enterprise. The Free-Net is funded by grants and donations from corporations, foundations, private citizens and governments. VIFA was put into operation in November of 1992 with one phone line. Anybody could log on, but for full access, a person had to download an application form, fill it out and mail it to VIFA.

The single phone line was soon swamped, making it virtually impossible to log on. Overtures were made to businesses, and soon VIFA had 8 lines due to their generosity. As of January 1996, there are 40 lines and it is difficult to log on evenings and weekends. More lines are planned, with some funding coming from users who voluntarily take out a membership for $25. This gives a registered user one vote in the society. It is not necessary to take out a membership to use the Free-Net. Personally, I have not taken out a membership but I make a donation of $15 every three months. VIFA now has 15,000 registered users out of a Victoria population of 300,000 and has 40 phone lines, 7.8 gigabytes of hard drive space and is run on a Sun minicomputer.

## FULL ACCESS

If you need full Internet access, service from a commercial provider is needed. The Free-Net does not provide File Transfer Protocol (FTP) which allows the downloading of files, however you can accomplish the same thing by going to a file area on certain mainframes like the University of Michigan or HENSA. These sites, and many others, have areas for Atari files, both 8-bit and ST, with thousands of PD files available. They have servers which you invoke with Unix-style command-line instructions, which are worse than MS DOS: The server then locates the file you want, UUencodes it, and sends it to your e-mail address. You then capture it in your buffer and when you log off, un-UUencode it. UUencoding simply takes a file and turns it into ASCII text to enable it to be sent as e-mail. Un-UUencoding turns it back to its original form, whether it be a Basic file, machine language or any other form.

As far as I know, there are no Web Browsers for the 8-bit. You can still go to all the Web sites and home pages. You will see all the text but wherever there is a picture, you will see [Graphic]. You can use all other areas of the Free-Net. If you have an ST, there is software available to get you graphics on the WWW.

For a graphical Web Browser on the ST, you can get two programs called STiK and CAB, which are freeware, although donations are suggested. The programs run on any ST, STe, Mega ST, TT030, Falcon030, Medusa040, Eagle, Direct30 and MagiCMac. They require that you have an Internet Service Provider that supports a SLIP connection. It will not work if the provider uses C SLIP or PPP. To get STiK and CAB look for WWW116 and download it. Other programs called MiNT and MINTNet reportedly give access to C SLIP and PPP servicers.

There is lots of talk on comp.sys.atari.st about users having trouble with these programs but it would appear to me these are due to poor documentation and user error. I would suggest sending questions about this software to the newsgroup comp.sys.atari.st (no dot at the end of st).

## GET ON-LINE!

There is a fascinating world out there on the Internet. You can find information on literally everything under the sun on it when you learn how to have your Atari use various search tools to do the looking. when you make it on-line, say hello by sending me e-mail at ua558@freenet.victoria.bc.ca (again no dot at the end of the address or it will not work).

***Contact Gordon Hooper at***

**ua558@freenet.victoria.bc.ca**

# THE ACCESSORY SHOP

## DISKS ... DISKS ...

### NIBBLER

Slinking through an unknown land Nibbler finds himself in an iridescent world of constant change. As time grows short his body grows longer and he risks running into himself. Can he survive?

CLOSEOUT PRICE 50p plus 50p p&p

### RAMPAGE

It's rough and tumble all the way as you control three incredibly nasty characters which bear a remarkable resemblance to King Kong, Godzilla and Wolf-man through an orgy of destruction in Chicago, New York and San Francisco. You have 150 days of destruction in 50 different cities. Time for some revenge!

CLOSEOUT PRICE £1.00 plus 70p p&p

### DRUID

Another classic in which, as Last of the Great Druids, you wander through the Dungeons of Acamantor with your mighty Golem by your side on a quest to destroy the four demon princes. As you delve deeper into the dungeons you will find chests containing spells of formidable power to aid you and Pentograms of Life which will heal and revitalise you. Excellent graphics and superb gameplay make this one of the best arcade games.

CLOSEOUT PRICE £1.00 plus 70p p&p

### MAXWELL'S DEMON

A simple yet challenging puzzle game which will get you totally addicted. Two games in one for double the fun

CLOSEOUT PRICE 50p plus 50p p&p

## CASSETTES

### LANCELOT

Level 9 recreate the time of wizards and the Knights of the Round Table in a three part adventure spanning the complete saga of the quest for the Holy Grail. Superbly researched and superbly written this text adventure is full of atmosphere and highly recommended.

CLOSEOUT PRICE £1.00 plus 90p p&p

### JUGGLE'S HOUSE

Another neat program for 3 to 6 year olds teachin g the concepts of inside, outside, upper and lower with Juggle's house and toyshop

CLOSEOUT PRICE £1.00 plus 70p p&p

### BATTALION COMMANDER

An exciting real-time tactical game with you as lieutenant colonel in charge of an entire armoured battalion. Choose from five different scenarios from a training mission against a Soviet tank battalion to tough assignments against the Chinese. You can adjust the relative strengths of your forces and the opposition and choose from 40 different maps and 5 different scenarios.

CLOSEOUT PRICE £1.00 plus 70p p&p

**Transfer these cassettes to disk!**



## LAST FEW ROMS

### POLE POSITION

ONLY 8 LEFT!

CLOSEOUT PRICE £1.50 plus 70p p&p

### BASIC CARTRIDGE

Some programs, especially some early public domain programs won't run on the XL/XE but they will if you plug in the old version of Basic.

CLOSEOUT PRICE £1.00 plus 70p p&p

### DESIGNERS PENCIL

A new programing language that is based around the creation of graphics screens. Difficult to explain but worth a go at this price!

CLOSEOUT PRICE £1.00 plus 70p p&p

## TRANSDISK IV

**is still available**

***Tells you how to transfer the software shown opposite - but you can do many more - over 300 programs successfully transferred!***

**TRANSDISK IS NOW ONLY £9.95**

**(previously £14.95)**

**AVAILABLE ONLY FROM PAGE 6**



# BARGAIN CASSETTES

## Your choice of

## any 5 cassettes for £1.50 plus 80p p&p

## any 10 cassettes for £2.00 plus £1.20 p&p

**180** ✓
Eight opponents, digitised speech, two player option. live joystick control, full matchplay scoring, superb playability.

**BOMB FUSION** ✓
A terrorist gang has planted bombs in the Nuclear Processing Plant and you have to go in to save the plant

**DESPATCH RIDER** ✓
Join the growing band of street demons who terrify the population of the big cities to get the parcels through.

**FEUD** ✓
You must "out-spell" your rival Wizard. Tread carefully as strange things can happen in this game of magic

**FOOTBALL MANAGER** ✓
Everything you might want in a managerial simulation. Far too many features to describe, but you won't be disappointed

**GHOSTBUSTERS** ✓
At this very moment hundreds of ghosts are making their way to the infamous spook central. Only you can save the world from disaster

**GUN LAW** ✓
Four months of bloody alien attacks have taken their toll. You are left to fight alone against ruthless and bloodthirsty killers with just a single machine gun

**HENRY'S HOUSE** ✓
Little Henry has shrunk and must navigate his way through the royal household to find the cure. Voted one of the all time great games

**INVASION** ✓
Mobilize your units and prepare for battle. This all action space conflict requires skill, strategy and tactics.

**KIKSTART** ✓
The ultimate off-road motorbike scramble. Guide your rider over the obstacles in this great game for 1 or 2 players

**LOS ANGELES SWAT** ✓
Rescue the hostages from the terrorist gangs holding out in West L.A. Clean up the streets. Blow away the bad guys

**MASTER CHESS** ✓
An excellent chess simulation with all the correct moves, various openings, in fact everything to keep you challenged

**MILK RACE** ✓
Cycling 1,000 miles is no mean feat - and you could end up feeling pretty exhausted by the time you've finished

**MR DIG** ✓
An old favourite in which Mr Dig has to dig for hidden food supplies in the 'Meanie' territory below ground.

**NINJA** ✓
Blasts the belt off all other martial arts games! It says on the inlay! Someone sure reckons this is the best punching, kicking, ducking and diving game of all

**ON CUE** ✓
A challenging real life simulation which combines Pool and Snooker on the same cassette. An absolute must for both enthusiasts and beginners alike.

**PANTHER** ✓
Save the last humans on Xenon. Take your ground attack ship through this 3D scrolling mega shoot-em-up with great graphics and unbelievable soundtrack

**PENGON** ✓
Can you save Penguin Willy from the ferocious mutant sea lions? Stun them by knocking them against the walls or crush them with sliding ice blocks

**PLASTRON** ✓
Take your place in a small band of pirates out to steal fossil fuels from the biggest corporation in the galaxy.

**PROTECTOR** ✓
Assigned to the US Army Helicopter Training School your aim is to become the best chopper pilot in the West

**REVENGE II** ✓
The Mutated 90 foot high, laser spitting death camels have rebelled against their captors the Zzyaxians and are out for revenge! A Jeff Minter classic

**ROCKFORD** ✓
The only true arcade version of the classic game Boulderdash. Explore 4 levels on each of 5 different worlds

**SIDEWINDER II**
It is 27 years since the final battle of the war with the aliens. All this is about to change. Step aboard your craft to defend mankind in this space blast

**SPEED HAWK**
Defend the ring worlds of your solar system from space pirates. Another of the great space games

**SPEED ZONE** ✓
Enter the Speedzone in a frantic defence against alien forces. A survey ship comes under attack and your "Starfire" class attack craft is launched

**STAR RAIDERS** ✓
What more can be said. Probably the best computer game in the world - ever!

**TAIL OF BETA LYRAE** ✓
Our A-Z of Atari Software series says "The ultimate 'Scramble' clone with superb graphics and music."

**TWILIGHT WORLD** ✓
Equipped with the latest in anti-gravity pods and Laser weaponry, battle your way through each of eleven dungeons

**UNIVERSAL HERO** ✓
Only seven seconds left to save the planet! Universal Hero has to save his skin and everybody else's by finding bits to repair a shuttle

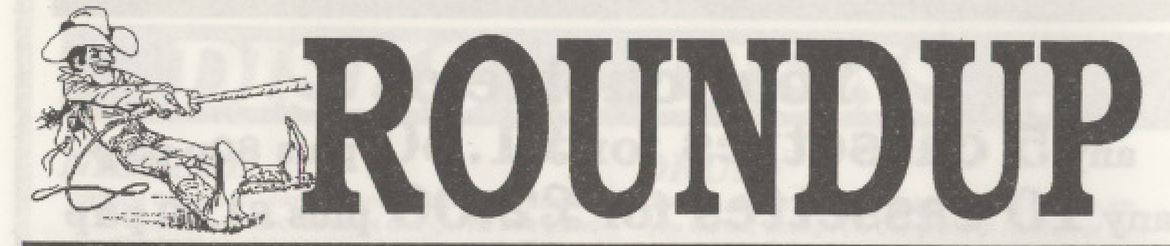
***✓ TRANSDISK IV shows you how to transfer these to disk!***

**ORDER ITEMS FROM THE ACCESSORY SHOP WITH THE ORDER FORM ENCLOSED WITH THIS ISSUE OR WRITE TO**

**PAGE 6, P.O. BOX 54, STAFFORD, ST16 1DR**

**TELEPHONE ORDERS ACCEPTED ON 01785 241153 USING ACCESS OR VISA**

# ST PUBLIC DOMAIN




*Les Ellingham has told me that he has no plans to add further titles to the current Page 6 ST Library catalogue. Although disappointing, this news is certainly understandable due to his ever increasing workload and the steady decline in orders for ST public domain software.*

*It is not all bad news though. Les has received letters asking for his ST support to continue. He will therefore continue to offer the current catalogue of over 1000 disks. I will continue to review these disks in ST Public Domain Roundup.*

*If you can, order some disks from the ST Library or simply write to Les offering your support. Perhaps we can persuade him to add the occasional new title when he comes across anything outstanding. New additions could be listed in the ST section of New Atari User. Give Les your thoughts on this subject.*

***by Stuart Murray***

Desktop Disc menu Edit menu

*Browsing through file.*

Holder: 1 | Number: 1 | Title: Untitled

| | | | | | | |
|---|---|---|---|---|---|---|
| Men | Women | Children | Babies | O.A.P.'s | Workers | Glamour |
| At home | Abroad | Holidays | Xmastime | Weddings | Parties | Sports |
| Schools | Music | Theatre | Events | Fairs | Exhibits | Crafts |
| Seaside | Rivers | Lakes | Hills | Trees | Farms | Parks |
| Cities | Buildings | Interiors | Shops | Adverts | Tools | Machinery |
| Transport | Bus/vans | 2-wheelrs | Aircraft | Ships | Boats | Railways |
| Horsedrwn | Cars | Military | Weaponry | Amusing | Signs | Hobbies |
| Animals | Pets | 'Cute' | Wildlife | Birds | Insects | Fish |
| Weather | Sunshine | Sunset | Night | Snow | Rain | Misty |
| Drought | Floods | Lightning | Clouds | Fire | Plants | Flowers |
| Furniture | Utensils | Food | Drink | Glass | Metal | Toys |
| Portrait | Landscape | Abstract | Reflected | Lowlight | Backlit | Flash |
| Dbl-xposd | Softfocus | Wideangle | Long-expo | Hi-speed | Close-up | Negative |
| Print | Slide | Colour | B/white | 35mm | 110/disc | Lgeformat |

00001 < < < > > > Exit!

## PHOTO FILER

If, like me, you have hundreds of photographs which have not been catalogued in photo albums then this program is definitely for you.

**PHOTO FILER** is an easy to use database for cataloguing your collection of photographs, slides or negatives. All you have to do is assign a number to each 'holder' (album, envelope, shoe box, etc.), number each photograph within the holder and enter this data into Photo Filer along with a title and the categories into which the photograph falls. There are 98 categories included with everything from men, women and children to music, cars, cute, abstract, portrait, pets, colour, black and white, 35mm, 110/disc and lots more. After selecting which categories the photograph falls into and saving the data to disk it is then a simple matter of searching by type to find the location of any photograph. Identification is given by holder/reference number and title.

Photo Filer is a simple program which could prove useful to many people - if only to save you the hassle of looking through pile after pile of photographs to recall memories of a past holiday.



## SPOT IT!

Spot It! is a unique implementation of that old puzzle favourite 'Spot the Difference'. There are eleven pairs of detailed pictures on the disk, each set presented together on the screen in a colourful, cartoon style. You use the mouse pointer to identify ten differences in the right-hand picture. These differences can be tiny due to the detail involved. Parents would call this game good for observational skills. I call it fun for all ages. It is a simple concept which has converted well to computer format. Check it out.

## REFLEXOR

This is another Arkanoid clone, although with a slight change to the gameplay. Basically you must shoot a hole through your opponent's wall whilst trying to stop his shots hitting your own. You can either play against a friend or the computer. One player plays bottom up, the other top down. So you want a review? Well, how about slow, jerky and virtually unplayable. Add to this a nasty screen flicker, a diabolical degree of slowdown and an exceedingly stupid computer player and you have Reflexor. This game is crap - avoid it at all costs!

## CRACK ART

The Germans brought us this shareware art package a few years back. It is now widely regarded as one of the best non-commercial packages available. There is a 20-page manual on the disk translated from the German original.

Crack Art is bursting at the seams with features, all available from a menu screen: Draw, Erase, Point, Line, K-Line, Rays, Frame, Circle, Airbrush, Box, Disc, Polygon, Spline (curves), Smear (smears pixels), Outline, Fill, Mirror and Text are the main drawing features. Add to these a number of special features such as up to nine work screens, pre-defined colour palettes, picture and block manipulation, colour animation, etc. and you have a top class product which deserves a place in any ST software collection. If you have an interest in computer art then Crack Art is for you. It is a real cracker which matches many commercial packages.

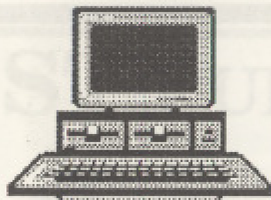
*Try as I might I couldn't get screen dumps from any of these programs except PHOTO FILER which is graphically the least interesting. Typical!*

*CRACK ART is a real joy to see but it saves in its own format and does not appear to be able to save in Degas of Neo format which could be a drawback.*

*Les Ellingham*

## ROUNDUP RATINGS:

| | |
|---|---|
| **ST1000 PHOTO FILER** | **79%** |
| **ST994 SPOT IT!** | **75%** |
| **ST958 REFLEXOR** | **10%** |
| **ST707 CRACK ART** | **94%** |




# JOURNEY INTO CYBERSPACE

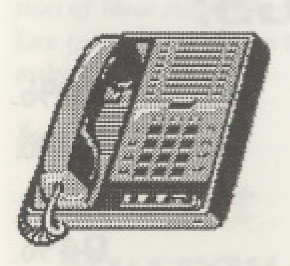
# COME IN TO THE WEB

***John S Davison explores the Internet in an attempt to explain the wonders of a new electronic world***

The next port of call on our cyberspatial trek currently represents THE killer application on the Internet. I'm talking about the "World Wide Web" (a.k.a. WWW, W3, or "the Web"), which is currently growing at a rate that's almost beyond belief.

Like many of the applications on the Internet the WWW was originally designed as an academic/research tool. It started life at CERN, the European Particle Physics Laboratory, as a means of easily accessing and viewing large quantities of interrelated research papers and other documents stored in computer files. The idea caught on and underwent dramatic development, and today it's used in all sorts of different (and unlikely) environments.

So what's all the fuss about? Well, the Web now represents the first easy-to-use point and click graphical environment for finding your way around the Internet. In addition it has built-in multimedia support, which means that as well as text it can also easily handle graphics, sound, and video data - all integrated into the same environment. In concept it's a bit like the Fastext teletext system fitted to many television sets, where you press coloured buttons on the controller to follow colour coded links to the information you want to access. But, it's much more versatile and handles far more than simple text data.



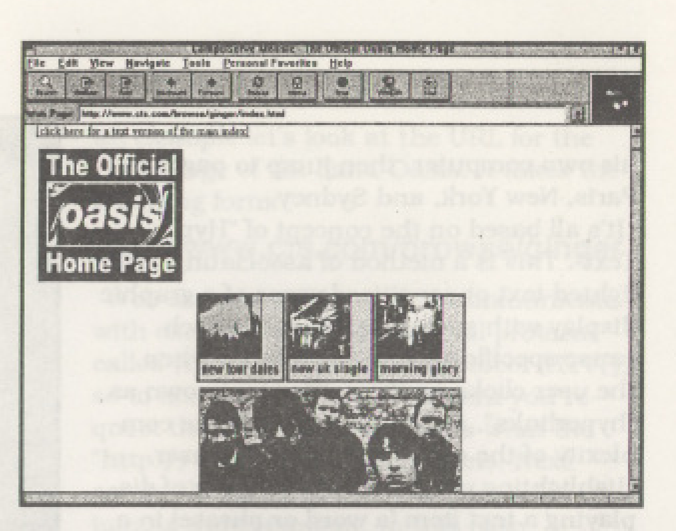


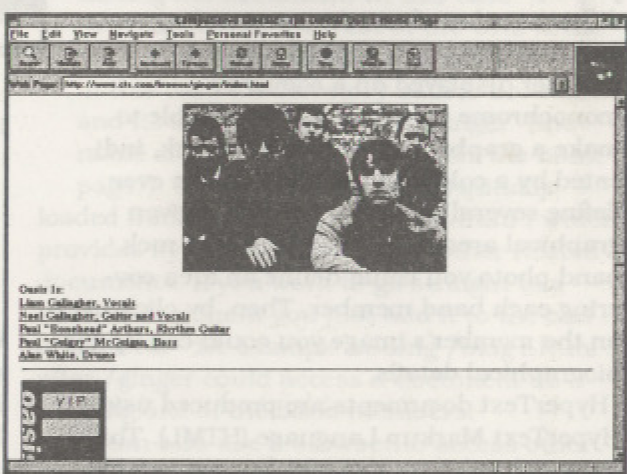


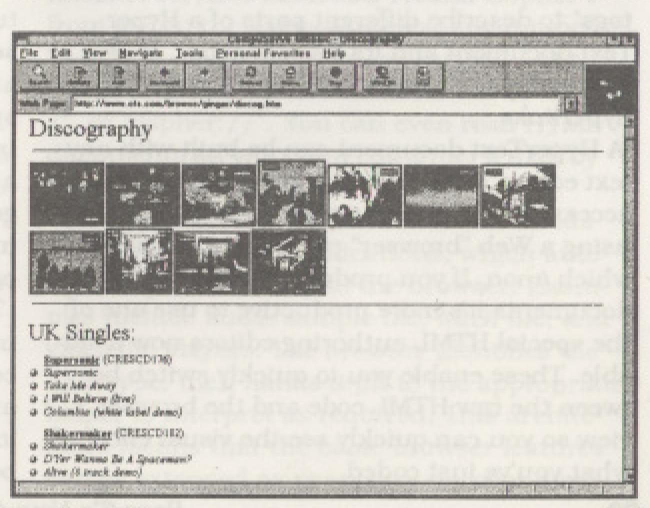
## CLICKING ALL OVER THE WORLD

For example, a rock band might set up its own WWW "Home Page". This is the name given to a collection of data relating to a given topic stored in a particular place on the Web, and provides a starting point for "Web surfing" activities. It could display text details about the band, graphics showing the band's logo, and photographs of the band members. It could also have clickable areas to trigger the playing of a track from the band's latest CD, or perhaps the showing of a clip from their latest video. It might also have controls to transfer the user to a different part of the home page containing additional data, or more significantly, take him to a completely different home page - perhaps one set up by the band's fan club.

The amazing thing about this is that subsequent pages can be on the same computer, on a different computer in the same building, in the next town, at the other end of the country, or on the other side of the planet. To the user there is NO apparent difference - he can jump from one to another ("surf the Web") with a single mouse click. In the space of a few seconds he could be looking at interlinked pages on



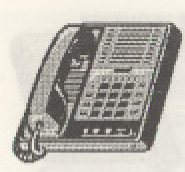




***Oasis home page - contains information about the band, photos, audio and video clips***

***Oasis home page - hyperlinks to details on band members***

***Oasis home page - hyperlinks to details of the bands records***



his own computer, then jump to pages in Paris, New York, and Sydney.

It's all based on the concept of "HyperText". This is a method of associating highlighted text or sensitised areas of a graphic display with special commands which cause specific actions to be taken when the user clicks on them. They're known as "hyperlinks", and effectively hide the complexity of the commands from the user. Highlighting usually takes the form of displaying a text item (a word or phrase) in a different colour from the surrounding text and underlining it, so it stands out whether displayed on a colour or monochrome screen. It's also possible to make a graphic item into a hyperlink, indicated by a coloured border. You can even define several hyperlinks within a given graphical area. For example, in the rock band photo you could define an area covering each band member. Then, by clicking on the member's image you could call up his biographical details.

HyperText documents are produced using HyperText Markup Language (HTML). This is a specialised language which uses "markup tags" to describe different parts of a HyperText document and its overall layout. It also associates the special commands with the hyperlinks.

A HyperText document can be built with any text editor, but to see how it will look when accessed via the Web you have to view it using a Web "browser" program, more of which anon. If you produce a lot of HyperText documents it's more productive to use one of the special HTML authoring editors now available. These enable you to quickly switch between the raw HTML code and the browser view so you can quickly see the visual effect of what you've just coded.




*CompuServe UK's new home page*

## FREE HOME PAGES

Home pages need somewhere to "live" on the Web, so users can access them. Some people set up their own Web server computer system and pay for this to be directly attached to the Internet. This is a very costly option, normally only done by companies and academic institutions. A cheaper option is to use a Web service provider, who already has a server attached to the Internet and will rent out space on it for other people's home pages. Incidentally, CompuServe have just launched a FREE home page service - subscribers now get 1MB disk space included free in their monthly subscription, for setting up their own personal home pages.

The server system accepts requests from users (generated by the embedded hyperlink commands in the document they're viewing) and acts on them appropriately, e.g. to download the next page of the HyperText document being viewed, or send an audio or video file for






*Gibson home page - hyperlinks to everything you wanted to know about Gibson guitars*

playback. The user end of the link is also handled by special software known as a Web browser. This usually has a graphical interface, and allows a user to easily enter the "Uniform Resource Locator" (URL) of a particular home page and then connect via the Internet with the server on which it lives. Once connected everything can be controlled via the browser's onscreen menus, toolbars, and the hyperlinks in the HTML document downloaded by the server.

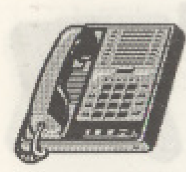
URL's are probably the most difficult thing to understand about the Web, as they look quite forbidding. A URL is a string of characters separated by colons and slashes, and is used to perform three functions. Firstly it identifies the Internet service you wish to use; secondly it indicates the Internet host computer where the service is located; and finally it indicates the resource you wish to access - usually expressed as a disk file system path to a HyperText document of some description. As an example let's look at the URL for the home page of the band Oasis. It takes the following form:

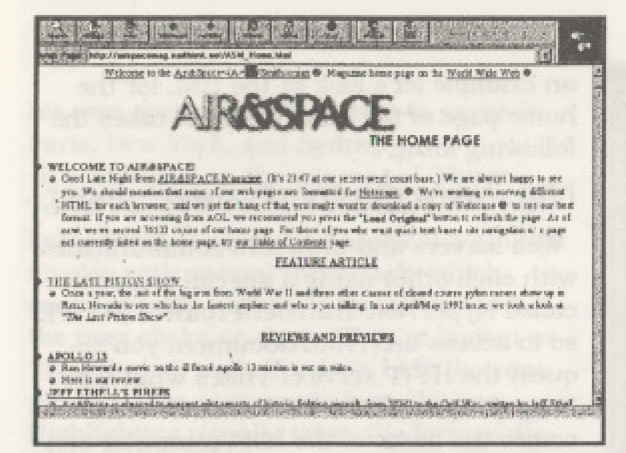
**http://www.cts.com/browse/ginger**

Web servers and browsers communicate with each other using a special protocol called HyperText Transfer Protocol (HTTP), so to access an HTML document you request the HTTP service. That's what the "http://" part of the URL means. Next comes the name of the host computer containing the document, in this case "www.cts.com". The rest of the URL is the path to the home page html documents - in this case the directory called "browse" and its subdirectory called "ginger" (the name of the guy who looks after the home page?). The default document initially loaded from here is called "index.html", which provides hyperlinks into all the other related documents. If you want to go straight to a specific document you just add it to the end of the URL - for example adding /biog1.html after /ginger could access a document describing one of the band members.

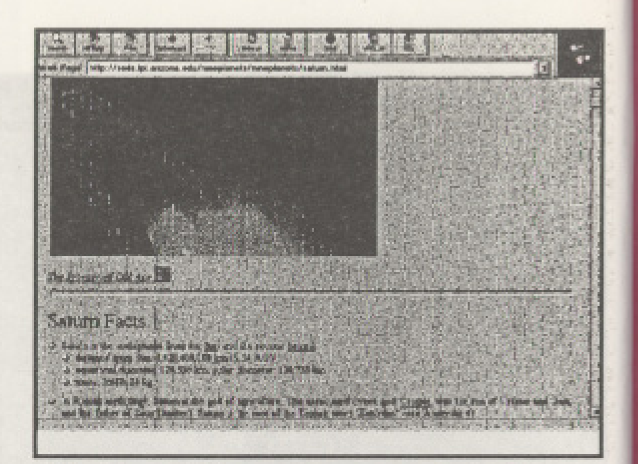
You can also use a browser to access other Internet services such as FTP and Gopher from within the same graphical environment. You get to them by replacing "http://" in the URL with other service names such as "ftp://", or "gopher://". You can even read HTML documents off your own computer's disk by using a service name of "file://".

Display of certain multimedia items needs additional "helper" applications, which automatically link directly to the browser. Examples include audio sample file, MIDI file, and video file players. The browser identifies the data type, then hands it off to the appropriate helper to interpret as required. This architecture means that the basic browser features can be extended as necessary to cover any





*Air & Space Magazine - for aircraft and spaceflight fans*



*Nine Planets home page - crammed with facts, figures, photos about our solar system*

data requiring special handling.

There are now more than a dozen different browsers available, the most popular being Netscape Navigator and NCSA Mosaic. As usual the Atari was left out in the cold - until very recently, that is. There are now a pair of programs called STIK and CAB, which work together to give ST users access to the Web. These were brought to my attention by Gordon Hooper, an NAU contributor resident in Canada. Gordon uses this software and kindly e-mailed details to me. However, I'm unable to try it out myself, as it uses a connection protocol known as SLIP and CompuServe, who provide my link to the Internet, only handle the PPP protocol at the moment. Foiled again! The good news is that Gordon has promised to write an article for NAU about these programs, hopefully to appear in the next issue. ●

## NAU INTERNET CONTACT LIST

I've started a list of NAU readers who would welcome e-mail from other Atari users. If you would like to be added to this list please drop me an e-mail note at the address below.

| | |
|---|---|
| John S Davison | 100256.1577@compuserve.com |
| Derek Fern | derek@dfern.demon.co.uk |
| Joel Goodwin | j.f.goodwin@reading.ac.uk |
| Gordon Hooper | ua558@freenet.victoria.bc.ca |
| Paul Rixon | p_rixon@oscarfab.demon.co.uk |
| Brad Rogers | brad@pianosa.demon.co.uk |
| Ann O'Driscoll | annod@iol.ie |
| Allan Palmer | 100644.1040@compuserve.com |
| Bryan Zillwood | b.j.zillwood@exeter.ac.uk |



## contact ... contact ... contact ...

### WANTED

**Urgently wanted, copy of Mapping The Atari (Revised), any books on machine code programming. Also any text adventures (disk or cassette). Tel. 01785 248020 and ask for Daniel**

### WANTED

**Original copy of PRINT-SHOP with manuals, Mapping The Atari, and any Atari ROMS in original packaging or with instructions. Call Chris on 01343 551379**

***This is all we received for Contact this issue - at least people have stopped selling their systems!***

### FREE TO SUBSCRIBERS

The CONTACT column is free of charge to subscribers who wish to sell their equipment or contact other readers. Space is limited so we request that entries be kept as short as possible. Extremely long entries may be heavily edited or ignored. Send your CONTACT notice on a separate sheet of paper (not as part of a letter) to:

**CONTACT, PAGE 6 PUBLISHING, STAFFORD, ST16 1DR**

FOR SALE ... WANTED ... PEN PALS ... ADVICE ... HELP

## PROGRAM LISTINGS

Certain program listings which are too long to include in the magazine may be obtained free of charge as printed listings to type in. All programs are, however, included on the Issue Disk which is available with each issue. Remember this disk also includes BONUS PROGRAMS which do not appear in the magazine. If you would like the type-in listings please write or telephone indicating which listings you require. Please note that there are not necessarily extra listings for every magazine.

***Write to LISTINGS, NEW ATARI USER, P.O. BOX 54, STAFFORD, ST16 1DR or telephone 01785 241153***